UNITED STATES AIR FORCE

SUMMER RESEARCH PROGRAM -- 1998

SUMMER RESEARCH EXTENSION PROGRAM FINAL REPORTS

VOLUME 3

ROME LABORATORY

RESEARCH & DEVELOPMENT LABORATORIES

5800 Uplander Way

Culver City, CA 90230-6608

Program Director, RDL
Gary Moore

Program Manager, AFOSR
Colonel Jan Cerveny

Program Manager, RDL
Scott Licoscos

Program Administrator, RDL
Johnetta Thompson

Program Administrator, RDL
Rebecca Kelly-Clemmons

Submitted to:

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH

Bolling Air Force Base

Washington, D.C.

December 1998

PREFACE

This volume is part of a four-volume set that summarizes the research of participants in the 1998 AFOSR Summer Research Extension Program (SREP). The current volume, Volume 1 of 5, presents the final reports of SREP participants at Armstrong Laboratory.

Reports presented in this volume are arranged alphabetically by author and are numbered consecutively -- e.g., 1-1, 1-2, 1-3; 2-1, 2-2, 2-3, with each series of reports preceded by a 35 page management summary. Reports in the five-volume set are organized as follows:

VOLUME	TITLE
1	Armstrong Research Laboratory
2	Phillips Research Laboratory
3	Rome Research Laboratory
4	Wright Research Laboratory
5	Air Logistics Center Arnold Engineering Development Center

REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-00-

0776

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December, 1998

3. REI

4. TITLE AND SUBTITLE
1998 Summer Research Program (SRP), Summer Research Extension Program (SREP), Final Report, Volume 3, Rome Laboratory

5. FUNDING NUMBERS
F49620-93-C-0063

6. AUTHOR(S)
Gary Moore

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Research & Development Laboratories (RDL)
5800 Uplander Way
Culver City, CA 90230-6608

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research (AFOSR)
801 N. Randolph St.
Arlington, VA 22203-1977

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The United States Air Force Summer Research Program (SRP) is designed to introduce university, college, and technical institute faculty members to Air Force research. This is accomplished by the faculty members, graduate students, and high school students being selected on a nationally advertised competitive basis during the summer intersession period to perform research at Air Force Research Laboratory (AFRL) Technical Directorates and Air Force Air Logistics Centers (ALC). AFOSR also offers its research associates (faculty only) an opportunity, under the Summer Research Extension Program (SREP), to continue their AFOSR-sponsored research at their home institutions through the award of research grants. This volume consists of a listing of the participants for the SREP and the technical report from each participant working at the AF Rome Laboratory.

14. SUBJECT TERMS
Air Force Research, Air Force, Engineering, Laboratories, Reports, Summer, Universities, Faculty, Graduate Student, High School Student

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements.***

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	**PR** - Project
G - Grant	**TA** - Task
PE - Program Element	**WU** - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with....; Trans. of....; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE - See authorities.
NASA - See Handbook NHB 2200.2.
NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.
DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
Leave blank.
NASA - Leave blank.
NTIS -

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self- explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

1998 SREP Final Technical Report Table of Contents

Armstrong Research Laboratory

Volume 1

1998 SREP Final Technical Report Table of Contents

Phillips Research Laboratory

Volume 2

1998 SREP Final Technical Report Table of Contents

Rome Research Laboratory

Volume 3

1998 SREP Final Technical Report Table of Contents

Wright Research Laboratory

Volume 4

1998 SREP Final Technical Report Table of Contents

Volume 5

1998 SUMMER RESEARCH EXTENSION PROGRAM (SREP) MANAGEMENT REPORT

1.0 BACKGROUND

Under the provisions of Air Force Office of Scientific Research (AFOSR) contract F49620-90-C-0076, September 1990, Research & Development Laboratories (RDL), an 8(a) contractor in Culver City, CA, manages AFOSR's Summer Research Program. This report is issued in partial fulfillment of that contract (CLIN 0003AC).

The Summer Research Extension Program (SREP) is one of four programs AFOSR manages under the Summer Research Program. The Summer Faculty Research Program (SFRP) and the Graduate Student Research Program (GSRP) place college-level research associates in Air Force research laboratories around the United States for 8 to 12 weeks of research with Air Force scientists. The High School Apprenticeship Program (HSAP) is the fourth element of the Summer Research Program, allowing promising mathematics and science students to spend two months of their summer vacations working at Air Force laboratories within commuting distance from their homes.

SFRP associates and exceptional GSRP associates are encouraged, at the end of their summer tours, to write proposals to extend their summer research during the following calendar year at their home institutions. AFOSR provides funds adequate to pay for SREP subcontracts. In addition, AFOSR has traditionally provided further funding, when available, to pay for additional SREP proposals, including those submitted by associates from Historically Black Colleges and Universities (HBCUs) and Minority Institutions (MIs). Finally, laboratories may transfer internal funds to AFOSR to fund additional SREPs. Ultimately the laboratories inform RDL of their SREP choices, RDL gets AFOSR approval, and RDL forwards a subcontract to the institution where the SREP associate is employed. The subcontract (see Appendix 1 for a sample) cites the SREP associate as the principal investigator and requires submission of a report at the end of the subcontract period.

Institutions are encouraged to share costs of the SREP research, and many do so. The most common cost-sharing arrangement is reduction in the overhead, fringes, or administrative charges institutions would normally add on to the principal investigator's or research associate's labor. Some institutions also provide other support (e.g., computer run time, administrative assistance, facilities and equipment or research assistants) at reduced or no cost.

When RDL receives the signed subcontract, we fund the effort initially by providing 90% of the subcontract amount to the institution (normally $18,000 for a $20,000 SREP). When we receive the end-of-research report, we evaluate it administratively and send a copy to the laboratory for a technical evaluation. When the laboratory notifies us the SREP report is acceptable, we release the remaining funds to the institution.

2.0 THE 1998 SREP PROGRAM

SELECTION DATA: A total of 490 faculty members (SFRP Associates) and 202 graduate students (GSRP associates) applied to participate in the 1998 Summer Research Program. From these applicants 188 SFRPs and 98 GSRPs were selected. The education level of those selected was as follows:

1997 SRP Associates, by Degree			
SFRP		GSRP	
PHD	MS	MS	BS
184	6	2	53

Of the participants in the 1997 Summer Research Program 90 percent of SFRPs and 13 percent of GSRPs submitted proposals for the SREP. One undred and thirty-two proposals from SFRPs and seventeen from GSRPs were selected for funding, which equates to a selection rate of 54% of the SFRP proposals and of 34% for GSRP proposals.

1998 SREP: Proposals Submitted vs. Proposals Selected			
	Summer 1997 Participants	Submitted SREP Proposals	SREPs Funded
SFRP	188	132	20
GSRP	98	17	4
TOTAL	286	149	24

The funding was provided as follows:

Contractual slots funded by AFOSR		18
Laboratory funded		22
	Total	40

Twelve HBCU/MI associates from the 1997 summer program submitted SREP proposals; six were selected (none were lab-funded; all were funded by additional AFOSR funds).

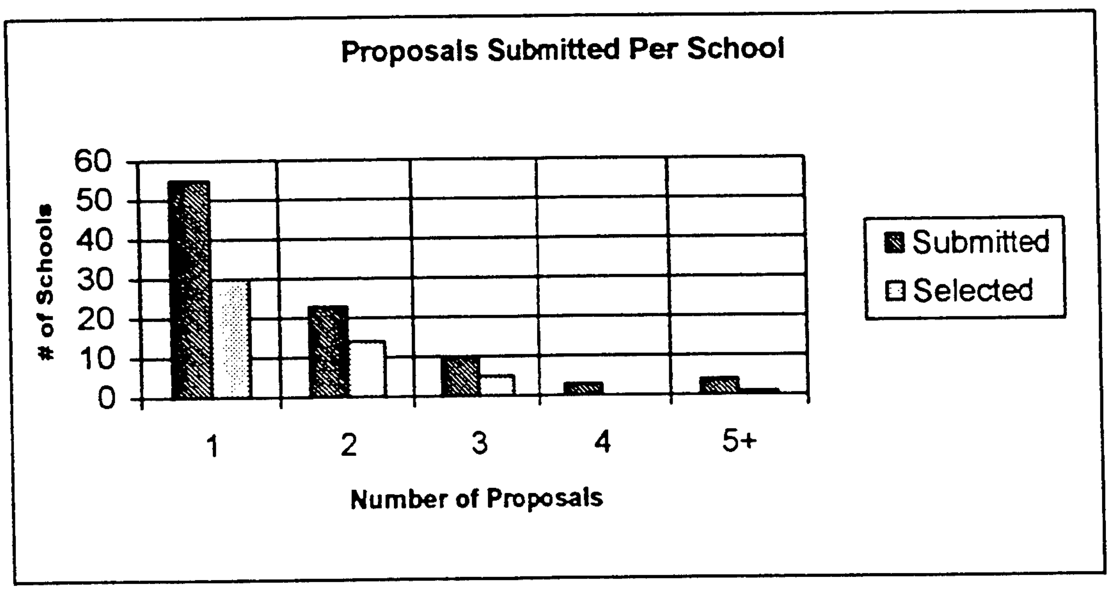
Proposals Submitted and Selected, by Laboratory		
	Applied	Selected
Armstrong Research Site	9	3
Air Logistic Centers	31	5
Arnold Engineering Development Center	2	1
Phillips Research Site	30	10
Rome Research Site	29	12
Wilford Hall Medical Center	1	0
Wright Research Site	47	9
TOTAL	149	40

Note: Armstrong Research Site funded 1 SREP; Phillips Research Site funded 6; Rome Research Site funded 9; Wright Research Site funded 6.

The 125 1997 Summer Research Program participants represented 60 institutions.

Institutions Represented on the 1997 SRP and 1998 SREP		
Number of schools represented in the Summer 97 Program	Number of schools represented in submitted proposals	Number of schools represented in Funded Proposals
125	110	55

Thirty schools had more than one participant submitting proposals.



The selection rate for the 65 schools submitting 1 proposal (68%) was better than those submitting 2 proposals (61%), 3 proposals (50%), 4 proposals (0%) or 5+ proposals (25%). The 4 schools that submitted 5+ proposals accounted for 30 (15%) of the 149 proposals submitted.

Of the 149 proposals submitted, 130 offered institution cost sharing. Of the funded proposals which offered cost sharing, the minimum cost share was $3046.00, the maximum was $39,261.00 with an average cost share of $11,069.21.

Proposals and Institution Cost Sharing		
	Proposals Submitted	Proposals Funded
With cost sharing	117	32
Without cost sharing	32	8
Total	149	40

The SREP participants were residents of 31 different states. Number of states represented at each laboratory were:

States Represented, by Proposals Submitted/Selected per Laboratory		
	Proposals Submitted	Proposals Funded
Armstrong Laboratory	31	5
Air Logistic Centers	9	3
Arnold Engineering Development Center	2	1
Phillips Laboratory	30	10
Rome Laboratory	29	12
Wilford Hall Medical Center	1	0
Wright Laboratory	47	9

Nine of the 1997 SREP Principal Investigators also participated in the 1998 SREP.

ADMINISTRATIVE EVALUATION: The administrative quality of the SREP associates' final reports was satisfactory. Most complied with the formatting and other instructions provided to them by RDL. Thirty-seveb final reports have been received and are included in this report. The subcontracts were funded by $992,855.00 of Air Force money. Institution cost sharing totaled $354.215.00.

TECHNICAL EVALUATION: The form used for the technical evaluation is provided as Appendix 2. Thirty-five evaluation reports were received. Participants by laboratory versus evaluations submitted is shown below:

	Participants	Evaluations	Percent
Armstrong Laboratory	5	4	95.2
Air Logistic Centers	3	3	100
Arnold Engineering Development Center	1	1	100
Phillips Laboratory	10	10	100
Rome Laboratory	12	12	100
Wright Laboratory	9	5	91.9
Total	40	35	95.0

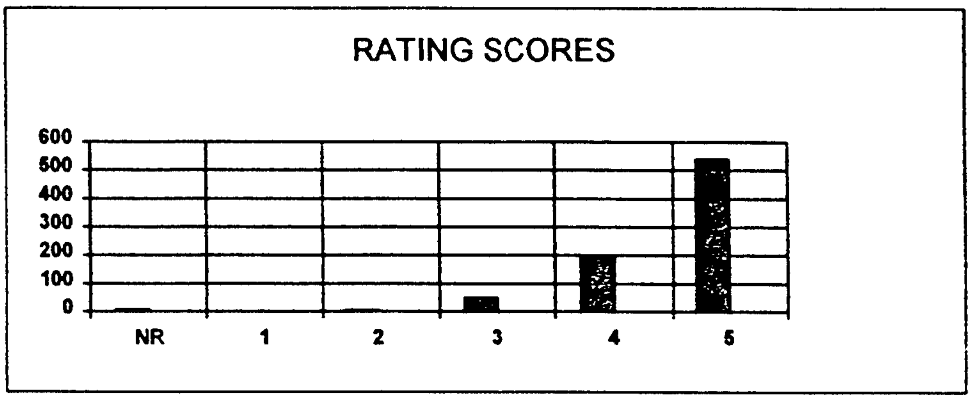
Notes:

1: Research on four of the final reports was incomplete as of press time so there aren't any technical evaluations on them to process, yet. Percent complete is based upon 20/21=95.2%

2: One technical evaluation was not completed because one of the final reports was incomplete as of press time. Percent complete is based upon 18/18=100%

The number of evaluations submitted for the 1998 SREP (95.0%) shows a marked improvement over the 1997 SREP submittals (65%).

PROGRAM EVALUATION: Each laboratory focal point evaluated ten areas (see Appendix 2) with a rating from one (lowest) to five (highest). The distribution of ratings was as follows:

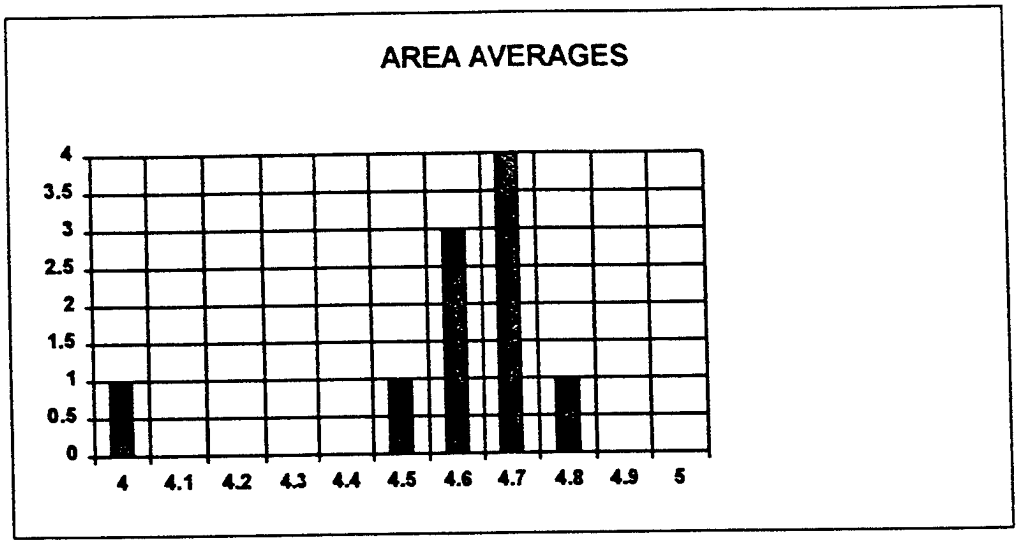


Rating	Not Rated	1	2	3	4	5
# Responses	7	1	7	62 (6%)	226 (25%)	617 (67%)

The 8 low ratings (one 1 and seven 2's) were for question 5 (one 2) "The USAF should continue to pursue the research in this SREP report" and question 10 (one 1 and six 2's) "The one-year period for complete SREP research is about right", in addition over 30% of the threes (20 of 62) were for question ten. The average rating by question was:

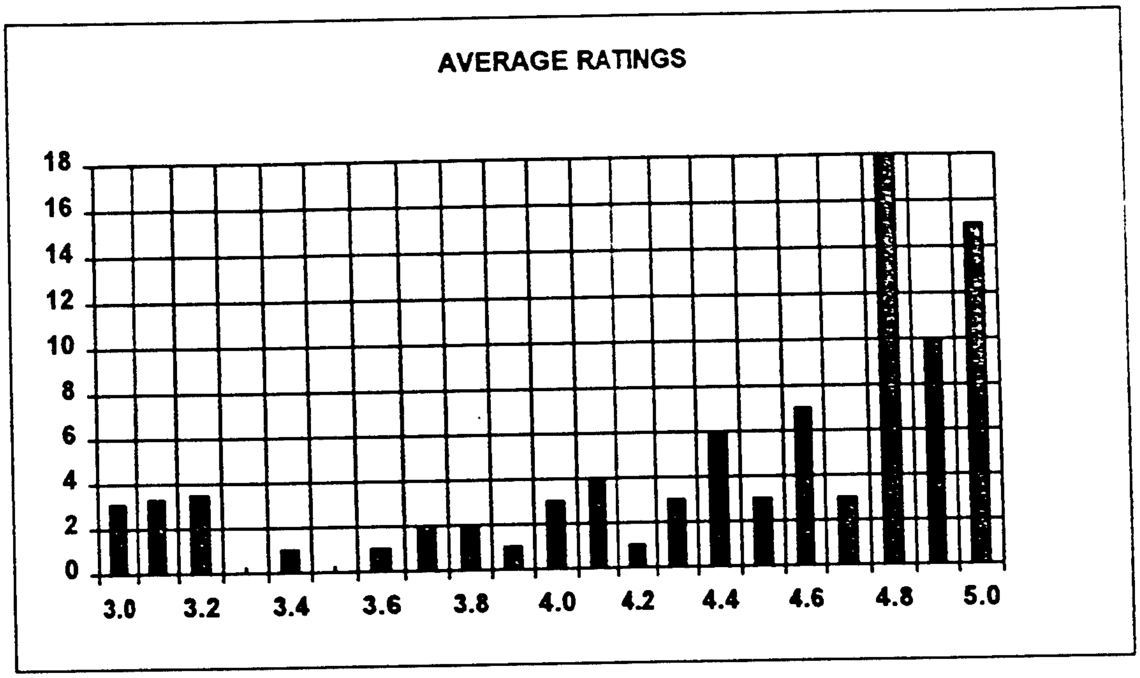
Question	1	2	3	4	5	6	7	8	9	10
Average	4.6	4.6	4.7	4.7	4.6	4.7	4.8	4.5	4.6	4.0

The distribution of the averages was:



Area 10 "the one-year period for complete SREP research is about right" had the lowest average rating (4.1). The overall average across all factors was 4.6 with a small sample standard deviation of 0.2. The average rating for area 10 (4.1) is approximately three sigma lower than the overall average (4.6) indicating that a significant number of the evaluators feel that a period of other than one year should be available for complete SREP research.

The average ratings ranged from 3.4 to 5.0. The overall average for those reports that were evaluated was 4.6. Since the distribution of the ratings is not a normal distribution the average of 4.6 is misleading. In fact over half of the reports received an average rating of 4.8 or higher. The distribution of the average report ratings is as shown:



It is clear from the high ratings that the laboratories place a high value on AFOSR's Summer Research Extension Programs.

3.0 SUBCONTRACTS SUMMARY

Table 1 provides a summary of the SREP subcontracts. The individual reports are published in volumes as shown:

Laboratory	Volume
Armstrong Research Site	1
Arnold Engineering Development Center	5
Air Logistic Centers	5
Phillips Research Site	2
Rome Research Site	3
Wright Research Site	4

SREP SUB-CONTRACT DATA

Report Author / Author's University	Author's Degree	Sponsoring Lab	Performance Period	Contract Amount	Univ. Cost Share
Chubb , Gerald	PhD	AL/HR	01/01/98 12/31/98	$25000.00	$0.00
Industrial Engineering	98-0829	Scoring Pilot Performance of Basic Flight Manuevers			
Ohio State University, Columbus, OH					
Foy , Brent	PhD	AL/OE	01/01/98 12/31/98	$25000.00	$11278.00
Medical Physics	98-0828	Development & Validation of a Physiologically-Based Kinetic Model of Perfused			
Wright State University, Dayton, OH					
Lance , Charles	PhD	AL/HR	01/01/98 12/31/98	$24989.00	$0.00
Psychology	98-0842	Extension of Job Performance Measurement Tech to the Development of a Prototype			
Univ of Georgia Res Foundation, Athens, GA					
Woehr , David	PhD	AL/HR	01/01/98 12/31/98	$25000.00	$11508.00
Department of Psychology	98-0802	Validation of The Multidimensional work ethic profile (MWEP) as a screening too			
Texas A & M Univ-College Station, College					
Collins , Frank	PhD	AEDC/E	01/01/98 12/31/98	$25000.00	$16104.00
Mechanical Engineering	98-0807	Monte Carlo Computation of SpeciesSepaaration by a Conical Skimmer in Hypersonic			
Tennessee Univ Space Institute, Tullahoma, TN					
Whaley , Paul	PhD	ALC/OC	01/01/98 12/31/98	$23351.00	$3046.00
Mechanical Engineering	98-0820	Probabilistic Analysis of Residual Strength in Corroded and Uncorroded Aging Air			
Oklahoma Christian Univ of Science & Art,					
Balas , Mark	PhD	PL/SX	01/01/98 12/31/98	$25000.00	$0.00
Applied Math	98-0816	Non-Linear Adaptive Control for a Precision Deployable Structure with White ligh			
Univ of Colorado at Boulder, Boulder, CO					
Duric , Neb	PhD	PL/LI	01/01/98 12/31/98	$25000.00	$5777.00
Astrophysics	98-0808	Image Recovery Using Phase Diversity			
University of New Mexico, Albuquerque, NM					
Hanson , George	PhD	PL/WS	01/01/98 12/31/98	$25000.00	$23250.00
Electrical Engineering	98-0811	Perturbation Analysis of the Natural Frequencies Targets in Inhomogeneous Media			
Univ of Wisconsin - Milwaukee, Milwaukee, WI					
Jeffs , Brian	PhD	PL/LI	01/01/98 12/31/98	$25000.00	$19177.00
Electrical Engineering	98-0813	Bayesian Restoration of Space object Images From Adaptive Optics Data with unkno			
Brigham Young University, Provo, UT					
Kar , Aravinda	PhD	PL/LI	01/01/98 12/31/98	$25000.00	$5414.00
Engineering	98-0812	Effects of Vapor-Plasma Layer on Thick-Section Cutting and Calculation of Modes			
University of Central Florida, Orlando, FL					
Leo , Donald	PhD	PL/VT	01/01/98 09/30/98	$24964.00	$9628.00
Mechanical & Aerospace	98-0810	Adaptive vibration suppression for autonomous Control Systems			
University of Toledo, Toledo, OH					
Liu , Hanli	PhD	PL/LI	01/01/98 12/31/98	$25000.00	$11000.00
Physics	98-0814	Continuous-Wave Approach to 3-D Imaging through Turbid media w/a Single Planar M			
Univ of Texas at Arlington, Arlington, TX					
Bienfang , Joshua	BS	PL/LI	01/01/98 12/31/98	$24994.00	$0.00
Physics	98-0815	Optical Clocks Based on Diode Lasers			
University of New Mexico, Albuquerque, NM					
Paulson , Eric	BS	PL/RK	01/01/98 12/31/98	$25000.00	$7794.00
Engineering/Physics	98-0837	Optimization & Analysis of a Waverider Vehicle for Global Spaceplane Trajectorie			
Univ of Colorado at Boulder, Boulder, CO					

SREP SUB-CONTRACT DATA

Report Author Author's University	Author's Degree	Sponsoring Lab	Performance Period		Contract Amount	Univ. Cost Share
Stephens II , Kenneth University of North Texas, Denton, TX	MA 98-0809	PL/WS	01/01/98	12/31/98	$25000.00	$16764.00
		Simulation of an Explosively Formed Fuse Using MACH 2				
Barjaktarovic , Milica Electrical Engineering Wilkes University, Wilkes Barre, PA	PhD 98-0824	RL/IW	01/01/98	12/31/98	$24976.00	$3158.00
		Specification and Verification of SDN.701 MSP Functions and Missi Crypto Functic				
Batalama , Stella EE SUNY Buffalo, Buffalo, NY	PhD 98-0823	RL/C3	01/01/98	12/31/98	$25000.00	$5600.00
		Robust Spread Spectrum Communications:Adaptive Interference Mitigation Technique				
Bourbakis , Nikolaos Computer Science & Engr SUNY Binghamton, Binghamton, NY	PhD 98-0832	RL/IR	01/01/98	12/31/98	$25000.00	$22723.00
		hierarchical-Adaptive Image Segmentation				
Dasigi , Venugopala Computer Science Southern Polytechnic State Univ, Marietta, GA	PhD 98-0830	RL/C3	01/01/98	12/31/98	$25000.00	$4000.00
		Information Fusion w/Multiple Feature Extractors for automatic Text Classificati				
Eckert , Richard Physics SUNY Binghamton, Binghamton, NY	PhD 98-0825	RL/C3	01/01/98	12/31/98	$25000.00	$39261.00
		The Interactive Learning Wall; A PC-Based, Deployable Data Wall for Use in a Co				
Lin , Kuo-Chi Aerospace Engineering University of Central Florida, Orlando, FL	PhD 98-0822	RL/IR	01/01/98	12/31/98	$25000.00	$0.00
		Web-Based Distributed Simulation				
Pados , Dimitrios Dept. of Electrical /Computer Eng. State Univ. of New York Buffalo, Buffalo, NY	PhD 98-0818	RL/OC	01/01/98	12/31/98	$25000.00	$5600.00
		Adaptive Array Radars and Joint Space-Time Auxiliary Verctor Filtering				
Panda , Brajendra Computer Science University of North Dakota, Grand Forks, ND	PhD 98-0821	RL/CA	01/01/98	12/31/98	$25000.00	$7113.00
		Information Warfare: Design of an Efficient Log Management Method to Aid In Dat				
Pittarelli , Michael Systems Science SUNY OF Tech Utica, Utica, NY	PhD 98-0827	RL/C3	01/01/98	12/31/98	$24998.00	$0.00
		Complexity of Detecting and content-driven methods for resolving database incons				
Schmalz , Mark Dept of Computer & Info Science University of Florida, Gainesville, FL	PhD 98-0831	RL/IR	01/01/98	12/31/98	$24619.00	$0.00
		Errors Inherent in 3D Target Reconstruction from Multiple Airborne Images				
Ye , Nong Industrial Engineering Arizona State University, Tempe, AZ	PhD 98-0826	RL/CA	01/01/98	12/31/98	$25000.00	$5000.00
		Model-Based Assessment of Campaign Plan Performance under Uncertainty				
Bradley , Parker Physics Syracuse University, Syracuse, NY	BS 98-0834	RL/IR	01/01/98	12/31/98	$25000.00	$0.00
		Development of User-Friendly Comp Environment for Blind Source Separation Studie				
Kumar , Devendra Computer Science CUNY-City College, New York, NY	PhD 98-0805	ALC/SA	01/01/98	12/31/98	$25000.00	$11362.00
		Further Development of a Simpler, Multiversion concurrency Control Protocol for				
Chow , Joe Mechanical Engineering Florida International Univ, Miami, FL	PhD 98-0806	ALC/W	01/01/98	12/31/98	$25000.00	$5360.00
		An Automated 3-D Surface Model Creation Module for Laser Scanned Point Data				

SREP SUB-CONTRACT DATA

Report Author / Author's University	Author's Degree	Sponsoring Lab	Performance Period		Contract Amount	Univ. Cost Share
Beecken , Brian **Physics** **Bethel College, St. Paul, MN**	**PhD** 98-0804	**WL/MN**	**01/01/98**	**12/31/98**	**$19986.00**	**$3997.00**
		Development of a statistical Model predicting the impact of a scene projector's				
Beggs , John **Electrical Engineering** **Mississippi State University, Mississippi State,**	**PhD** 98-0817	**WL/FI**	**01/01/98**	**12/31/98**	**$25000.00**	**$25174.00**
		Implementation of an Optimization Algorithm in Electromagnentics for Radar Absor				
Bhatnagar , Raj **Computer Science** **University of Cincinnati, Cincinnati, OH**	**PhD** 98-0819	**WL/AA**	**01/01/98**	**09/30/98**	**$25000.00**	**$17488.00**
		Analysis of Intra-Class Variability & synthetic Target Models for Use in ATR				
Blaisdell , Gregory **Mechanical Engineering** **Purdue University, West Lafayette, IN**	**PhD** 98-0839	**WL/FI**	**01/01/98**	**12/31/98**	**$25000.00**	**$11844.00**
		Validation of a Large Eddy Simulation Code & Development of Commuting Filters				
Douglass , John **Zoology** **University of Arizona, Tucson, AZ**	**PhD** 98-0803	**WL/MN**	**01/01/98**	**12/31/98**	**$25000.00**	**$3719.00**
		Roles of Matched Filtering and Coarse in Insect Visual Processing				
Hosford , William **Mettalurgy** **Univ of Michigan, Ann Arbor, MI**	**PHD** 98-0840	**WL/MN**	**01/01/98**	**12/31/98**	**$25000.00**	**$5000.00**
		Prediction of Compression Textures in Tantalum Using a Pencil-Glide Computer Mod				
Pan , Yi **Computer Science** **University of Dayton, Dayton, OH**	**PhD** 98-0838	**WL/FI**	**01/01/98**	**12/31/98**	**$25000.00**	**$9486.00**
		Parallelization of Time-Dependent Maxwell Equations Using High Perform. Fortran				
Pochiraju , Kishore **Mechanical Engineering** **Stevens Inst of Technology, Hoboken, NJ**	**PhD** 98-0833	**WL/ML**	**01/01/98**	**12/31/98**	**$25000.00**	**$9625.00**
		A Hybrid Variational-Asymptotic Method for the Analysis of MicroMechanical Damag				
Shtessel , Yuri **Electrical Engineering** **Univ of Alabama at Huntsville, Huntsville, AL**	**PhD** 98-0841	**WL/FI**	**01/01/98**	**12/31/98**	**$25000.00**	**$4969.00**
		Continuous Sliding Mode Control Approach for Addressing actutor Deflection and				
Starzyk , Janusz **Electrical Engineering** **Ohio University, Athens, OH**	**PhD** 98-0801	**WL/AA**	**01/01/98**	**12/31/98**	**$24978.00**	**$12996.00**
		Feature Selection for Automatic Target Recognition:Mutual Info & Stat Tech				

APPENDIX 1:

SAMPLE SREP SUBCONTRACT

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
1998 SUMMER RESEARCH EXTENSION PROGRAM
SUBCONTRACT 98-0812

BETWEEN

Research & Development Laboratories
5800 Uplander Way
Culver City, CA 90230-6608

AND

University of Central Florida
Office of Sponsored Research/ Admin#423
4000 Central Florida Blvd.
Orlando, FL 32816-0150

REFERENCE: Summer Research Extension Program Proposal 97-0018

Start Date: 01/01/98 End Date: 12/31/98

Proposal Amount: $25000.0
Proposal Title:
Effects of Vapor-Plasma Layer on Thick-Section Cutting and Calculation of Modes

(1) **PRINCIPAL INVESTIGATOR:**
DR Aravinda Kar
CREOL
University of Central Florida
Orlando, FL 32816-2700

(2) **UNITED STATES AFOSR CONTRACT NUMBER: F49620-93-C-0063**

(3) **CATALOG OF FEDERAL DOMESTIC ASSISTANCE NUMBER (CFDA): 12.800**
PROJECT TITLE: AIR FORCE DEFENCE RESEARCH SOURCES PROGRAM

(4) **ATTACHMENTS**
1 **REPORT OF INVENTIONS AND SUBCONTRACT**
2 **CONTRACT CLAUSES**
3 **FINAL REPORT INSTRUCTIONS**

*** SIGN SREP SUBCONTRACT AND RETURN TO RDL ***

1. BACKGROUND: Research & Development Laboratories (RDL) is under contract (F49620-93-C-0063) to the United States Air Force to administer the Summer Research Program (SRP), sponsored by the Air Force Office of Scientific Research (AFOSR), Bolling Air Force Base, D.C. Under the SRP, a selected number of college faculty members and graduate students spend part of the summer conducting research in Air Force laboratories. After completion of the summer tour participants may submit, through their home institutions, proposals for follow-on research. The follow-on research is known as the Summer Research Extension Program (SREP). Approximately 61 SREP proposals annually will be selected by the Air Force for funding of up to $25,000; shared funding by the academic institution is encouraged. SREP efforts selected for funding are administered by RDL through subcontracts with the institutions. This subcontract represents an agreement between RDL and the institution herein designated in Section 5 below.

2. RDL PAYMENTS: RDL will provide the following payments to SREP institutions:

- 80 percent of the negotiated SREP dollar amount at the start of the SREP research period.
- The remainder of the funds within 30 days after receipt at RDL of the acceptable written final report for the SREP research.

3. INSTITUTION'S RESPONSIBILITIES: As a subcontractor to RDL, the institution designated on the title page will:

a. Assure that the research performed and the resources utilized adhere to those defined in the SREP proposal.

b. Provide the level and amounts of institutional support specified in the SREP proposal..

c. Notify RDL as soon as possible, but not later than 30 days, of any changes in 3a or 3b above, or any change to the assignment or amount of participation of the Principal Investigator designated on the title page.

d. Assure that the research is completed and the final report is delivered to RDL not later than twelve months from the effective date of this subcontract, but no later than December 31. 1998. The effective date of the subcontract is one week after the date that the institution's contracting representative signs this subcontract, but no later than January 15, 1998.

e. Assure that the final report is submitted in accordance with Attachment 3.

f. Agree that any release of information relating to this subcontract (news releases, articles, manuscripts, brochures, advertisements, still and motion pictures, speeches, trade associations meetings, symposia, etc.) will include a statement that the project or effort depicted was or is sponsored by: Air Force Office of Scientific Research, Bolling AFB, D.C.

g. Notify RDL of inventions or patents claimed as the result of this research as specified in Attachment 1.

h. RDL is required by the prime contract to flow down patent rights and technical data requirements to this subcontract. Attachment 2 to this subcontract contains a list of contract clauses incorporated by reference in the prime contract.

4. All notices to RDL shall be addressed to:

RDL AFOSR Program Office
5800 Uplander Way
Culver City, CA 90230-6609

5. By their signatures below, the parties agree to provisions of this subcontract.

______________________ Abe Sopher RDL Contracts Manager	______________________ Signature of Institution Contracting Official
	______________________ Typed/Printed Name
______________________ Date	______________________ Title
	______________________ Institution
	______________________ Date/Phone

ATTACHMENT 2
CONTRACT CLAUSES

This contract incorporates by reference the following clauses of the Federal Acquisition Regulations (FAR), with the same force and effect as if they were given in full text. Upon request, the Contracting Officer or RDL will make their full text available (FAR 52.252-2).

FAR CLAUSES	TITLE AND DATE
52.202-1	DEFINITIONS
52.203-3	GRATUITIES
52.203-5	COVENANT AGAINST CONTINGENT FEES
52.203-6	RESTRICTIONS ON SUBCONTRACTOR SALES TO THE GOVERNMENT
52.203-7	ANTI-KICKBACK PROCEDURES
52.203-8	CANCELLATION, RECISSION, AND RECOVERY OF FUNDS FOR ILLEGAL OR IMPROPER ACTIVITY
52.203-10	PRICE OR FEE ADJUSTMENT FOR ILLEGAL OR IMPROPER ACTIVITY
52.203-12	LIMITATION ON PAYMENTS TO INFLUENCE CERTAIN FEDERAL TRANSACTIONS
52.204-2	SECURITY REQUIREMENTS
52.209-6	PROTECTING THE GOVERNMENT'S INTEREST WHEN SUBCONTRACTING WITH CONTRACTORS DEBARRED, SUSPENDED, OR PROPOSED FOR DEBARMENT
52.212-8	DEFENSE PRIORITY AND ALLOCATION REQUIREMENTS
52.215-2	AUDIT AND RECORDS - NEGOTIATION
52.215-10	PRICE REDUCTION FOR DEFECTIVE COST OR PRICING DATA

52.215-12	SUBCONTRACTOR COST OR PRICING DATA
52.215-14	INTEGRITY OF UNIT PRICES
52.215-8	ORDER OF PRECEDENCE
52.215.18	REVERSION OR ADJUSTMENT OF PLANS FOR POSTRETIREMENT BENEFITS OTHER THAN PENSIONS
52.222-3	CONVICT LABOR
52.222-26	EQUAL OPPORTUNITY
52.222-35	AFFIRMATIVE ACTION FOR SPECIAL DISABLED AND VIETNAM ERA VETERANS
52.222-36	AFFIRMATIVE ACTION FOR HANDICAPPED WORKERS
52.222-37	EMPLOYMENT REPORTS ON SPECIAL DISABLED VETERAN AND VETERANS OF THE VIETNAM ERA
52.223-2	CLEAN AIR AND WATER
52.223-6	DRUG-FREE WORKPLACE
52.224-1	PRIVACY ACT NOTIFICATION
52.224-2	PRIVACY ACT
52.225-13	RESTRICTIONS ON CONTRACTING WITH SANCTIONED PERSONS
52.227-1	ALT. I - AUTHORIZATION AND CONSENT
52.227-2	NOTICE AND ASSISTANCE REGARDING PATIENT AND COPYRIGHT INFRINGEMENT

52.227-10	FILING OF PATENT APPLICATIONS - CLASSIFIED SUBJECT MATTER
52.227-11	PATENT RIGHTS - RETENTION BY THE CONTRACTOR (SHORT FORM)
52.228-7	INSURANCE - LIABILITY TO THIRD PERSONS
52.230-5	COST ACCOUNTING STANDARDS - EDUCATIONAL INSTRUCTIONS
52.232-23	ALT. I - ASSIGNMENT OF CLAIMS
52.233-1	DISPUTES
52.233-3	ALT. I - PROTEST AFTER AWARD
52.237-3	CONTINUITY OF SERVICES
52.246-25	LIMITATION OF LIABILITY - SERVICES
52.247-63	PREFERENCE FOR U.S. - FLAG AIR CARRIERS
52.249-5	TERMINATION FOR CONVENIENCE OF THE GOVERNMENT (EDUCATIONAL AND OTHER NONPROFIT INSTITUTIONS)
52.249-14	EXCUSABLE DELAYS
52.251-1	GOVERNMENT SUPPLY SOURCES

DOD FAR CLAUSES	DESCRIPTION
252.203-7001	SPECIAL PROHIBITION ON EMPLOYMENT
252.215-7000	PRICING ADJUSTMENTS
252.233-7004	DRUG FREE WORKPLACE (APPLIES TO SUBCONTRACTS WHERE THERE IS ACCESS TO CLASSIFIED INFORMATION)
252.225-7001	BUY AMERICAN ACT AND BALANCE OF PAYMENTS PROGRAM
252.225-7002	QUALIFYING COUNTRY SOURCES AS SUBCONTRACTS
252.227-7013	RIGHTS IN TECHNICAL DATA - NONCOMMERCIAL ITEMS
252.227-7030	TECHNICAL DATA - WITHOLDING PAYMENT
252.227-7037	VALIDATION OF RESTRICTIVE MARKINGS ON TECHNICAL DATA
252.231-7000	SUPPLEMENTAL COST PRINCIPLES
252.232-7006	REDUCTIONS OR SUSPENSION OF CONTRACT PAYMENTS UPON FINDING OF FRAUD

APPENDIX 2:

SAMPLE TECHNICAL EVALUATION FORM

SUMMER RESEARCH EXTENSION PROGRAM
TECHNICAL EVALUATION

SREP No: 98-0810
Principal Investigator: DR Donald Leo
University of Toledo

Circle the rating level number, 1 (low) through 5 (high), you feel best evaluate each statement and return the completed form to RDL by fax of mail to:

RDL
Attn: SREP Tech Evals
5800 Uplander Way
Culver City, CA 90230-6608
(310)216-5940 or (800)677-1363

1.	This SREP report has a high level of technical merit.	1 2 3 4 5
2.	The SREP program is important to accomplishing the lab's mission.	1 2 3 4 5
3.	This SREP report accomplished what the associate's proposal promised.	1 2 3 4 5
4.	This SREP report addresses area(s) important to the USAF.	1 2 3 4 5
5.	The USAF should continue to pursue the research in this SREP report.	1 2 3 4 5
6.	The USAF should maintain research relationships with this SREP associate.	1 2 3 4 5
7.	The money spent on this SREP effort was well worth it.	1 2 3 4 5
8.	This SREP report is well organized and well written.	1 2 3 4 5
9.	I'll be eager to be a focal point for summer and SREP associates in the future.	1 2 3 4 5
10.	The one-year period for complete SREP research is about right.	1 2 3 4 5

---

11. If you could change any one thing about the SREP program, what would you change:

____

12. What do you definitely NOT change about the SREP program?

____

PLEASE USE THE BACK FOR ANY OTHER COMMENTS

Laboratory Phillips Laboratory
Lab Focal Point Capt Jeanne Sullivan
Office Symbol AFRL/VSDV Phone: (505)846-2069

CASE STUDY IN MODULAR SYSTEM DESIGN: FORTEZZA-MSP INTERFACE

Milica Barjaktarović
Assistant Professor
Department of Electrical and Computer Engineering

Wilkes University
Stark Learning Center
Wilkes-Barre PA 18766

Final Report for:
Summer Research Extension Program

Sponsored by:

Air Force Office of Scientific Research
Bolling AFB

and

Wilkes University

January 15, 1999

CASE STUDY IN MODULAR SYSTEM DESIGN: FORTEZZA-MSP INTERFACE

Milica Barjaktarović
Assistant Professor
Department of Electrical and Computer Engineering
Wilkes University
Wilkes-Barre PA 18766

Abstract

In this document we continue to demonstrate the process of modular design and step-wise system refinement. We use the case study of a real-life system, the Multilevel Information System Security Initiative (MISSI). MISSI is a National Security Agency (NSA) program. designed to send protected messages over unprotected networks such as Internet. MISSI employs cryptography by using a credit card sized Personal Computer Memory card Interface Association (PCMCIA) card called the FORTEZZA Crypto Card.

MISSI relies on X.500. X.400, SDN.701-901 standards, as well as FORTEZZA documentation. Our research so far was focusing on specifying various parts of MISSI sender and receiver. in order to demonstrate modular design and stepwise refinement. We formally defined various requirements either directly specified or indirectly implied in the standards and documents describing MISSI, such as: message structure. MISSI access and forwarding policy, sender, and local cache. We assumed that the FORTEZZA card implements the necessary steps to carry out the given commands. In this report we take the specification into one more level of detail: we specify the interface between FORTEZZA card and application software, and the steps to be taken by FORTEZZA card. We used formal language called PROMELA, based on Hoare's CSP. We verified the model using automated model checker SPIN. developed by AT&T.

Contents

List of Figures

Chapter 1

Introduction

In the process of software design and testing, it is practical to rely on modular design and step-wise refinement in order to streamline and manage the process, especially if the system is secure and/or critical. We have used a real-life complex system to investigate how formal methods can assist in software design and testing.

Our case study is the Multilevel Information System Security Initiative (MISSI), a National Security Agency (NSA) program, designed to send protected messages over unprotected networks such as Internet [2]. MISSI relies on cryptography by using a credit card sized Personal Computer Memory card Interface Association (PCMCIA) card called the FORTEZZA Crypto Card (or "Card," for short).

FORTEZZA Card is not specific to MISSI system. As such, the Card is designed to run with any application which requires security services. The cryptographic hardware on the Card is accessed via high-level C functions, called the CI library of functions.

On the user's side, MISSI relies on SDN standards to invoke the proper security services. In other words, there is a software on the user's workstation, called User Agent (UA), which translates user requests into appropriate action. UA invokes high-level functions called MSP functions, according to the SDN standard.

Since FORTEZZA and SDN standards are two separate standards, we must be concerned with their interface. All ambiguities must be cleared. In particular, are there any omissions and/or errors which lead to security holes? For example, what happens when a user removes and inserts the card(s)?

From the engineering perspective, the two basic questions are:

- how do we design and build the interface?
- how do we know that it works correctly?

In this report, we will discuss the above questions.

This report is a continuation of our work presented in [3]. The goals of our work are:

- help the designers of complex systems such as MISSI comprehend and design the system in an organized way
- help the communication between teams designing and building various parts of the system
- eventually hand the verified formal specification to the programmers and builders: based on their input, change the specification, and repat this step.

We chose formal language PROMELA, based on Hoare's CSP, to specify and test MISSI receiving and forwarding.

Chapter 2 gives an overview of the formal tool we used, SPIN; Chapter 3 describes the MISSI application; Chapter 4 presents a detailed description of CI and MSP functions; Chapter 5 presents an example of research done so far and the results: and Chapter 8 presents the conclusion and discusses the future research. We have also included a brief discussion of cross-certificates in Chapter 7.

Chapter 2

Promela and SPIN

Promela is a language based on CSP and implemented in C. SPIN is the publicly distributed automated verification tool which supports Promela [6, 8, 7, 9].

Promela uses processes (we can think of them as "C functions executed concurrently.") Processes are executed by being called from another process using the `run` statement. The main process is called `init`.

Promela uses channels for communication. A channel is a FIFO queue of a certain capacity. For example, a channel which can store N byte-long messages is declared as:

```
chan channel = [N] of {byte}
```

Zero-capacity channels do not have capability to store messages, and are used for synchronization (i.e. handshaking.) The following syntax is used to append a message type `msg_type` with values `expr1` and `expr2` to channel `q`: `q!msg_type(expr1, expr2)`. `q?msg` means "receive message `msg` on channel `q`."

Promela has twelve types of statements: assertion, expression, selection, assignment, goto, break, repetition, atomic, send, receive, printf, and timeout. `skip` statement is analogous to C *continue* statement. Assertions, expressions and printf statements are similar to C.

Selection statement is similar to C *if* construct, except that Promela requires that one option be executed, and will deadlock if no option is executable. Repetition statement is similar to C *do* loop. It is executed repeatedly until a `break` statement is executed or a `goto` statement transfers control outside of the loop. Atomic sequence is indicated by the keyword `atomic` and allows a block of critical code to be executed without interruption.

SPIN can use `assert` statements, `never` claims and Linear Temporal Logic (LTL) statements for validation. Assert statements state properties of data at a given state. Never claims state properties of data at one or more states.

An `atomic` statement marks the specification section that is to be executed without interruptions: i.e. it is a semaphore protecting critical code, and is used for streamlining the model.

Chapter 3

Overview of MISSI Workstation

The overview of MISSI system is given in [5]. In this document we will give a brief overview of the MISSI workstation.

Each MISSI workstation may be connected to a FORTEZZA card reader. Each user authorized to use card reader services posses a FORTEZZA card. Cards are used for security functions, such as encryption and digital signatures. The Card is contacted via SDN.801 Message Security protocol. The process of sending out a message to the Internet involves using X.400 or RFC1521 message formatting mechanisms. X.400 is a suite of standards for message formatting and transfer. Each workstation has application software called User Agent (UA). which sends and receives messages.

Encryption and signatures require public and secret keys. A certificate is a data structure which includes each user's name and public keys. signed by the authority that issued the certificate. X.509 standard specifies the certification hierarchy and certificate and key management. Certificates are kept in a public depository called the Directory, specified by X.500 standard.

Workstations are equipped with either FORTEZZA Cards (F) or FORTEZZA Plus cards (FP), or they have no cards. FP cards handle Secret messages. F cards handle Sensitive But Unclassified messages. Users are certified by the Certification Authority (CA) operating at the CA workstation (CAW).

The abstract view of the interface between a MISSI workstation and FORTEZZA card consists of three interacting units: UA, MSP functions, and the Card. UA application process contacts the hardware on the Card through MSP application software. MSP application software represents a high-level interface to the Card and consists of a library of eight `msp_` functions. These functions call on the library of fifty one Crypto Interface (CI) `CI_` functions, which interface with the card device driver and eventually with the hardware. A card is inserted into a slot in a card reader. Each slot has a socket associated with it.

The interfaces are shown in Fig. 3.1. In Figure 3.2 we show the software components within a MISSI workstation and its connection to the FORTEZZA hardware. The details of the relationships between Card. Library, and sockets are outlined in section 4.1.

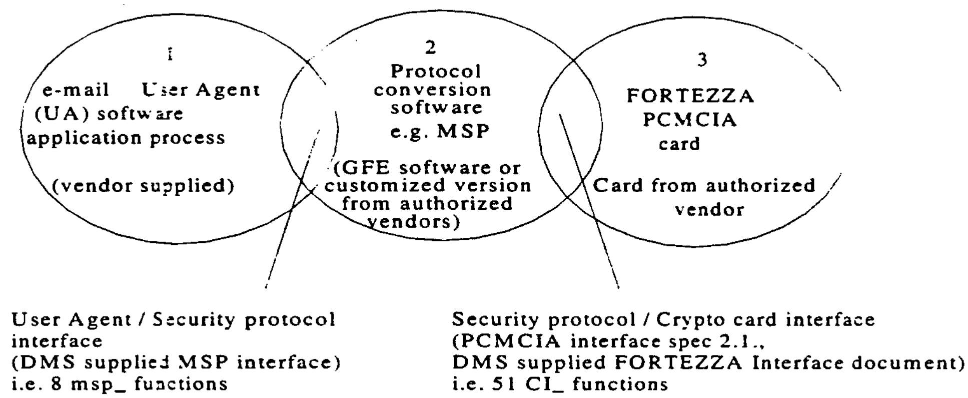


Figure 3.1: Abstract view of MISSI workstation /FORTEZZA card interface.

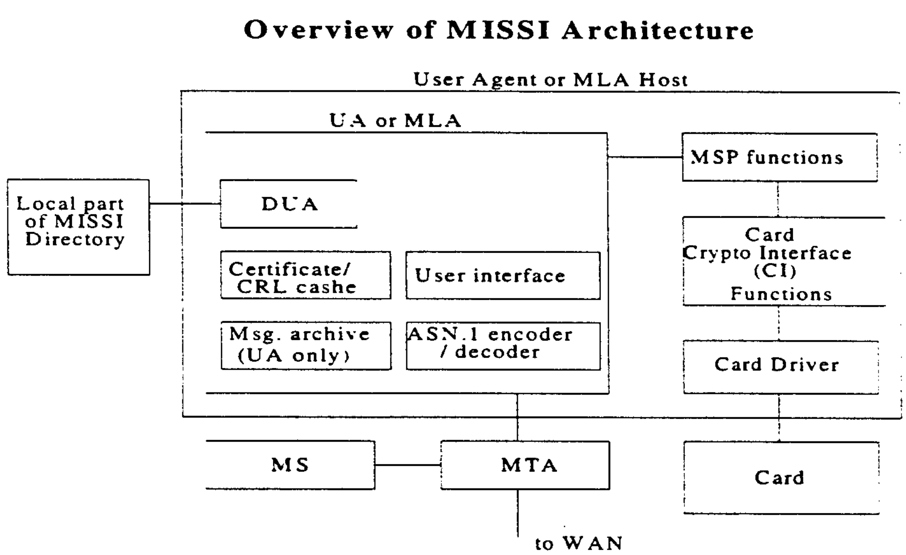


Figure 3.2: Local workstation components.

Chapter 4

MSP and CI Functions

In this chapter we will refer to Fig.3.2 and section 3.

msp_ functions are a high-level interface to CI_ functions. CI_ functions are grouped into a CI library, and executed using a Card. Card is inserted into a slot operated by a socket.

4.1 Definitions

We must more formally define various terminologies scattered throughout the MSP and CI documentation.

A Card is physically inserted into a socket. The socket has to be opened and selected in order for functions to be applied to the Card.

Each card is unlocked by only one PIN. which can be either User or SSO type. SSO-type PINs belong to SSO-enabled users, who have the power to load and unload information on a card. User-type PINs are used to log in users who do not have SSO privileges.

Each Card stores certificates. The certificates ordinarily belong to several personalities of the user who owns the PIN. It would be possible that several users know the PIN that unlocks a given card.

Each user supplies a PIN to a card, and eventually is prompted to choose one of the several personalities residing on the Card. Personalities can be changed any time during a session. without having to log out.

To log in a User, in case of non-SSO users, is accomplished by using `msp_login`. Logging in means to supply a PIN that unlocks the card. In this case, the Card is reset (explained below). The security functions on the Card are not available yet, until the user select a personality, which is beyond the domain of `msp_login`.

To reset a Card means to "zeroize all registers and common memory, and to log off User or SSO Enabled User" [10]. [1] p.25 defines reset as "clearing all internal registers."

To log off a User is not explicitly specified. For non-SSO users. we assumed that it means to change the state of Card to UserInitialized, so that the User must supply his PIN again and go through the logging on process, using msp_login; i.e. once a user logs in, he stays logged in until he removes the card or logs out. The meaning of logging off for a SSO enabled user is not explicitly specified either, and is not clear. This would be the topic of additional research.

Informally, the login algorithm is as follows (assuming that PIN is correct):

```
user requests login
MSP UA calls on msp_login
    a socket is opened or selected, card is inserted
    Card  is Reset, Card State = User Initialized
    user supplies PIN =>
         card is unlocked, Card State = Standby
```

After login, the user is obliged to select a personality using CI_Select in order to achieve the card state Ready.

More formally, msp_login algorithm is as follows:

```
CI_Initialize           /*initialize CI library*/
CI_Reset                /*reset card*/
CI_Open                 /*open socket*/
CI_GetConfiguration     /*check card's configuration*/
CI_CheckPIN             /*unlock card*/
for each certificate on the Card,
  build a list: (personality label,certificate index)
```

We found error in this algorithm. as CI_Open must happen before CI_Reset.

msp_clear specification is to "call the CI functions to clear all registers, terminate the Fortezza library, and close the Card socket " [1], p.25. This would indicate that msp_clear algorithm is as follows:

```
CI_Reset                /*reset card*/
CI_Terminate            /*terminate CI library*/
CI_Close                /*close socket, reset*/
```

However, the same page also states that "msp_clear logs the user off the Card by resetting it, thereby clearing all internal registers. The CI function called by msp_clear will also close the Card socket " [1], p.25. This would indicate that msp_clear algorithm is as follows:

```
CI_Close (CI_NULL_FLAG)  /*close socket, reset*/
```

We chose the last specification.

4.2 Card, CI Library and Socket States

CI Library can be either Initialized (i.e. ready for execution) or Uninitialized. CI_LIB_ALRDY_INIT is the flag that indicates if the library is already initialized or not.

Card can be in the following states:

Card State	Description
CI_PowerUp	self test, determine Card state
CI_Zeroize	zeroized
CI_Uninitialized	uninitialized, or zeroized and PIN was not entered; supply default SSO PIN
CI_Initialized	init. params loaded: change SSO PIN
CI_SSOInitialized	SSO PIN set; build or load certificates
CI_LAWInitialized	user certificates and params loaded: install user PIN
CI_UserInitialized	supply PIN to log on
CI_Standby	choose personality
CI_Ready	execute security functions

Socket can be in the following states, assuming that socket is powered up "permanently" (i.e. even if we power it down, we power it up immediatelly):

Socket states	Description
CardNotInserted	we defined this state. to complete the specs
SocketClosed	we defined this state. to complete the specs
CI_SocketUnlocked	socket is unlocked
CI_HoldLock	this application holds the lock to the socket
CI_SocketLocked	socket is locked and unavailable

We defined the composite states of the library and the socket:

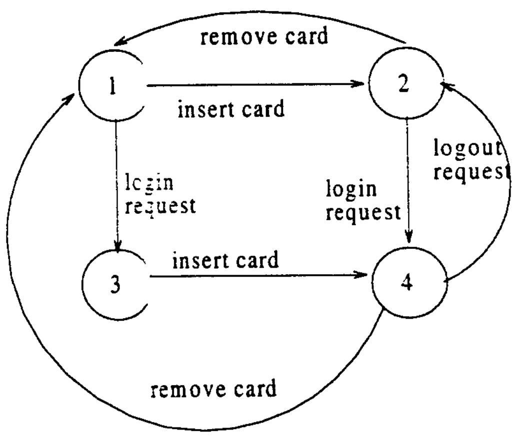
```
State1  CI_LIB_ALRDY_INIT == 0 && socketState[currentSocket] == SocketClosed

State2  CI_LIB_ALRDY_INIT != 0 && socketState[currentSocket] == SocketClosed

State3  CI_LIB_ALRDY_INIT != 0 && socketState[currentSocket] != SocketClosed
```

"Curent socket" refers to the socket currently selected. Once a socket is selected. either by `CI_Open` or `CI_Select`, all subsequent commands refer to that socket.

We also have to take into account what actions a user can execute based on the physical condition of the card. We define states as seen from the User's point of view:

User states	Description
in_State1	not logged in, card not inserted
in_State2	not logged in, card inserted
in_State3	logged in, card not inserted
in_State4	logged in, card inserted

These states are valid for every slot (i.e. socket) the User has access to. Therefore, we construct a state diagram for possible states of the Card and sockets and user action (Fig. 4.1). This state diagram is valid for one socket and needs to be activated for each socket separately.



1: In_State1: no card inserted, not logged in
2: In_State2: card inserted, not logged in
3: In_State3: card not inserted, started to log in
4: In_State4: card inserted, logged in

Figure 4.1: User Action Allowed

Chapter 5

SPIN Specification of CI-MSP Interface

This work illustrates a "divide and conquer" approach to specification and verification of a large, complex system like MISSI.

We model MISSI system as modules, where we abstract the modules which are less important by their input-output function. and focus on the modules of interest. Therefore. we specify and verify the entire system. by verifying the interaction between the components as well as the internal details of the components.

Currently, we have four different models of MISSI: [3] specifies a detailed model of MISSI sender with an abstraction of local cache and certificate verification; [4] also specifies the sender, but focuses on the local cache and certificate verification, and abstracts away the processes necessary to prepare a message for sending: [5] abstracts away the sending process. and focuses on the receiver and forwarding. In this model, we provide the finishing piece of MSP UA workstation, and specify the MSP-CI function interface.

5.1 Card Swapping and Context Switching

There are several possible scenarios when working with the cards:

- users are free to work with several cards and personalities at a time.
- we can have several MSP UAs working at the same time with one card reader (e.g. several windows opened)
- we can have several CI libraries at the same time.

In this work. we assume one CI library, one MSP UA, two sockets, and two cards.

"Hot swap" denotes removing one card and inserting another, in the middle of a session. "Context switching" denotes several MSP UAs working concurrently with one card reader. Therefore. if we use CCSs' Composition operator to denote concurrent behavior, our goal is to specify:

```
MSP_UA1 | MSP_UA2 | ...| MSP_UAn |
socket1 | socket2 | ... | socketm |
CI_library
```

We have specified hot swap support.

In our model, we have two sockets:

```
MSP_UA1 | socket1 | socket2| CI_library
```

[10], p.52, states: "Application should probably support different startup processes, perhaps one for "cold start" (system boot and reboot) and another for hot swap support (remove one card, insert another)."

Chapter 6

Results

We tested our model using a Linear Temporal Logic (LTL) formula, and an assert statement.

First we prove that, every time **msp_login** is executed correctly, the user's card used for login is in Standby state. We have done this proof for a card that belongs to a non-SSO user. SSO user behavior is not specified completely in the CI documentation, and needs some additional research.

```
msp_login!pin_type, pin, socketNo;
from_msp_login?rsp;
if
::(rsp==MSP_OK) -> skip; /*user will select personality*/
   atomic{
   logged_out = 0; /*user logged in*/
   called_CI_Reset = 0;
   }
  if
  ::(pin_type_that_passed==USER_PIN_TYPE) ->
    assert ( CardState[CardThatPassed] == CI_Standby);
    /*must verify this way because Cards can be removed and inserted*/
  ::(pin_type_that_passed == SSO_PIN_TYPE) -> skip;
  fi;
  sync!0;  /*synchronizing semaphore*/
  from_msp_ua!MSP_OK;
::(rsp!=MSP_OK) ->
 from_msp_ua!MSP_FAIL; error;
fi;
```

The synchronizing semaphore **sync** is used to assure that the test applies to the desired card. It is possible, after login, to insert and remove any card into the card reader slot. The synchronizing semaphore assures that we are referring to the card that was used for the last login process.

The next step is to prove that, every time a card is in Standby mode, i.e. a non-SSO user is logged in, the user stays logged in until card is removed or reset, or user loggs out. We prove the

LTL formula $\Box$(q U R), which means: "for all states, it holds that a q-interval either runs to the end of computation or is terminated by an r-state." We defined q and r as follows for socket0:

```
q = CardState[Card]==CI_Standby && socketContains[0]==Card
r = socketContains[0] == CardNotInserted || called_CI_Reset == 1 ||
    logged_out == 1
```

Similarly, we define the same tests for socket1, using socketContains[1].

SPIN uses different algorithms for exhaustive "guaranteed coverage" search or compressed "expected coverage" search. Expected coverage uses hashing techniques for state space compression and memory reduction. The compilation option DBITSTATE evokes exhaustive search. The compilation option COLLAPSE is one of the options that can be used for compression. One of the caveats is that running an LTL formula restricts the exhaustive run to the states matched by the formula.

Therefore, we ran our model in compressed search with LTL formula and in exhaustive search without it. Exhaustive search was done by substituting DBITSTATE option for COLLAPSE option. We ran our model as:

```
spin -a main.spin
gcc -DMEMCNT=28 -DCOLLAPSE -DSAFETY -o pan pan.c
pan -w23  -m300000
spin -t -p main.spin > look
```

The results without LTL formula are:

```
Depth=  104301 States=   1e+06 Transitions= 1.26712e+06 Memory= 14.388
Depth=  120996 States=   2e+06 Transitions= 2.74418e+06 Memory= 14.900
(Spin Version 3.2.0 -- 8 April 1998)
+ Partial Order Reduction

Bit statespace search for:
never-claim          - (none specified)
assertion violations +
cycle checks         - (disabled by -DSAFETY)
invalid endstates +

State-vector 200 byte, depth reached 120996, errors: 0
2.37169e+06 states, stored
1.0109e+06 states, matched
3.38259e+06 transitions (= stored+matched)
    9942 atomic steps
hash factor: 3.53698 (best coverage if >100)
(max size 2^23 states)

Stats on memory usage (in Megabytes):
```

```
483.824 equivalent memory usage for states (stored*(State-vector + overhead))
2.097  memory used for hash-array (-w23)
8.400  memory used for DFS stack (-m300000)
14.900 total actual memory usage

details about unreached states ...
```

The results with LTL formula are:

```
warning: never-claim + accept-labels requires -a flag to fully verify
warning: for p.o. reduction to be valid the never claim must be stutter-closed
(never claims generated from LTL formulae are stutter-closed)
Depth=   76228 States=   1e+06 Transitions= 1.26914e+06 Memory= 66.559
Depth=   96636 States=   2e+06 Transitions= 2.68848e+06 Memory= 91.033
Depth=   96636 States=   3e+06 Transitions= 4.10866e+06 Memory= 114.995
Depth=   96636 States=   4e+06 Transitions= 5.4948e+06 Memory= 139.059
Depth=   96636 States=   5e+06 Transitions= 6.90053e+06 Memory= 163.123
(Spin Version 3.2.0 -- 8 April 1998)
+ Partial Order Reduction
+ Compression

Full statespace search for:
never-claim          +
assertion violations + (if within scope of claim)
cycle checks         - (disabled by -DSAFETY)
invalid endstates - (disabled by never-claim)

State-vector 204 byte, depth reached 96636, errors: 0
5.65043e+06 states, stored
2.17159e+06 states, matched
7.82202e+06 transitions (= stored+matched)
   40986 atomic steps
hash conflicts: 1.29224e+06 (resolved)
(max size 2^23 states)

Stats on memory usage (in Megabytes):
1220.492 equivalent memory usage for states (stored*(State-vector + overhead))
135.869 actual memory usage for states (compression: 11.13%)
State-vector as stored = 12 byte + 12 byte overhead
33.554 memory used for hash-table (-w23)
7.200  memory used for DFS stack (-m300000)
178.687 total actual memory usage

nr of templates: [ globals procs chans ]
collapse counts: [ 1165 242 5064 276 3 43 2 ]

details about unreached states ...
```

Chapter 7

Cross Certification in X.509

The old version of SDN standard, SDN.701, and X.509 standard specified only certificates issued by a single Root certification authority (PAA).

The new SDN, SDN.801, specifies certificates issued by several PAAs. For example, each of the allied countries can have a PAA, issuing certificates fro that country. If someone from the USA wants to exchange secure mail with someone from, say, France, the certificates will be issued by different PAAs.

Some of the problems that we encounter in this new approach include:

- In the one-PAA system the PAA was the point of trust that the entire certificate validation was based on. In the case of cross-certificates, we need to find out how security can be preserved.
- The access rules, such as PRBAC, need to be specified in case of multiple PAAs. This task was started for one-PAA domains in [5]. Additional work needs to be done for multiple PAAs.

Let us assume that we have users User1 and User2. User1 is certified by CA1, which is certified by PCA1, which is certified by PAA1. User2 is certified by CA2, which is certified by PCA2, which is certified by PAA2. Users 1 and 2 can exchange secure mail iff PAA1 and PAA2 are mutually certified.

Scenario 1: User2 wants to send mail to User1. User2 would then find the certificate of User1 from the public Directory. In fact, this certificate will be signed by CAA1; which means that User2 will have to retrieve certificate of CAA1, signed by PCA1, and certificate of PCA1 signed by PAA1.

In the one-PAA system, the PAA's certificate was loaded on the user's Card. The whole certification procedure was based on this certificate being securely and accurately stored on the Card. User1 would have used this certificate to unsign the certificate of PAA1, then would use certificate of PAA1 to validate certificate of CA1, and finally use certificate of CA1 to validate User2's certificate.

In the multi-PAA case, User2 does not have certificate of PAA1 on his Card. Therefore, the validation process could be completed iff User2 can somehow obtain certificate of PAA1. We propose the following: *Each Card must be loaded with the certificate of the PAA(s) under which the Card's owner resides (let us call these PAAs "home PAAs"), and certificates of all other PAAs cross-certified by the "home PAAs."* For example, for the above example, Card of User2 would have been loaded with the certificate of PAA2 and certificate of PAA1 signed by PAA2.

Chapter 8

Conclusion

This work illustrates a "divide and conquer" approach to specification and verification of a large. complex system like MISSI.

We model MISSI system as modules. where we abstract the modules which are less important by their input-output function, and focus on the modules of interest. Therefore, we specify and verify the entire system, by verifying the interaction between the components as well as the internal details of the components.

Currently. we have four different models of MISSI: [3] specifies a detailed model of MISSI sender with an abstraction of local cache and certificate verification; [4] also specifies the sender, but focuses on the local cache and certificate verification, and abstracts away the processes necessary to prepare a message for sending; [5] abstracts away the sending process, and focuses on the receiver and forwarding. In this model, we provide the finishing piece of MSP UA workstation. and specify the MSP-CI function interface.

What we find is that specifications seem correct when done in English, and that the subtleties needed for final implementation can be recoved only with formal specification. We find that lack of formal documentation leaves many unspecified and conflicting details. The critical piece is always the interface between various modules.

Bibliography

[1] *Interface Control Document for the FORTEZZA Message Security Protocol Software.* http://www.armadillo.huntsville.al.us /Fortezza_docs/ obsolete.html, December 20 1994.

[2] *FORTEZZA Application Developers Guide.* Publicly available documents at
http://www.armadillo.huntsville.al.us, 14 July 1995.

[3] Milica Barjaktarović. Formal Specification and Verification of MISSI Architecture Using SPIN. Technical Report 4. AFOSR Summer Research Program, Rome Laboratory, Rome NY, September 1996.

[4] Milica Barjaktarović. Formal Specification and Verification of MISSI Local Cashe Using SPIN. Technical report, Rome Laboratory and ORA. Rome Laboratory, Rome NY, December 1996.

[5] Milica Barjaktarović. Formal Specification and Verification of MISSI Forwarding Policy Using SPIN. Technical report, Rome Laboratory. Rome Laboratory. Rome NY, May 1997.

[6] Gerard J. Holtzman. *Validation and Verification of Communication Protocols.* Prentice Hall, New York, 1989.

[7] Gerard J. Holzmann. Basic SPIN Manual. AT&T Bell Laboratories. Murray Hill, New Jersey, 079074.

[8] Gerard J. Holzmann. Design and Validation of Protocols: a Tutorial. *Computer Networks and ISDN Systems*, 25:981–1017, 1993.

[9] J.M. Troya P. Merino. Modeling and Verification of the ITU-T Multipoint Communication Service with SPIN. In *Proceedings of the 2nd International Workshop on the SPIN Verification System.* DIMACS, Rutgers University, New Brunswick. New Jersey, August 1996.

[[illegible]] Spyrus, 2814 Junction Ave, Suite 110, San Jose, CA. *FORTEZZA Application Implementors Guide*, 30 January 1995.

The State-of-the-Art in Formal Methods

Currently, there are more than 75 different formal methods (FMs) listed on the de-facto formal methods repository at the World Wide Web Virtual Library on Formal Methods, http://www.comlab.ox.ac.uk/archive/formal-methods/. This is a de-facto database of anything relating to formal methods, with entries mostly from the USA and Europe. Formal methods are in different stages of development, in a wide spectrum from formal languages with no tool support, to internationally standardized languages with tool support and industrial users. The field of formal methods is in a great flux and evolving rapidly, leaving research laboratories and making inroads into industrial practice.

One of the major issues in the technology transfer from academic research to industrial practice is how to make formal methods more available to practitioners. First, formal methods are not appealing to industry unless there are adequate tools that support the methods. Second, since real problems require diverse approaches and no single formal method is suitable for solving all classes of problems, it seems necessary to take advantage of different strengths of different formal methods. It would be highly desirable to integrate formal methods, especially those that have tool support.

We focus on formal tool integration as a practical step in a more widespread use of formal methods. In order to assess the latest developments in the field of formal methods and tools, we participated in several meetings and conferences. Our goal was to investigate possibilities for deriving a framework for integrating formal tools into a toolkit, either on its own, or as a part of existing toolkits.

We participated in the following events: two conferences focused on formal methods, one predominantly software oriented (WIFT'98), and another predominantly hardware oriented (FMCAD'98); a visit to a commercial research facility (SRI International); a research meeting (DARPA PI meeting); and the IEEE Computer Science Software Engineering Standards Committee (SESC) Formal Methods Planning Group. This wide variety of events exposed us to academic, industrial, and government formal methods communities, including tool makers, tool users, and funding agencies.

The overall impression is that industry is actively seeking new approaches that could assist in producing complex products while meeting short time-to-market deadlines. Industry is considering formal methods as a systematic approach to dealing with the overwhelming amount of information. CAD industry seems the most willing and already widely uses many lightweight formal tools such as model checkers and equivalence checkers. Telecommunications industry comes next. Some of the expressed needs include: more user-friendly, powerful and robust tools; more real-life applications; more infrastructure such as verified libraries; more publicity of success stories and available technologies; and more training.

Proposed Solutions

Based on our findings, we have constructed an outline of steps that can be taken to allow for integration of formal tools into existing industrial practice.

Step 1

The main questions to be answered are:

- What are the problems that users need solved?
- What kind of tools and features would be useful to the users?

To answer these questions, we must interview decision makers and cutting-edge researchers to find out what direction the field is moving in, and domain engineers to find out what the practicing engineers use and need in their daily work. In other words, we need to understand the current process flow and directions in which it is evolving, as well as the tools used. This task might be difficult to realize in practice, because commercial companies are very protective of their process flow as it contains trade secrets. Off-the-shelf tools are highly customized to fit a particular design flow. Companies that might be more willing to disclose their process flow are small and medium size companies.

More specific questions that need to be answered are:

- What application domain should the toolkit address?
  Since different application domains require different notations and features, it would be the best to develop a toolkit for one specific application. We narrow down the choices for relevant application domains to: telecommunications, CAD, and software industries. Telecommunications and CAD industries are the most mature application domains suitable for introduction of formal methods tools. Software industry in general is not expressing sufficient interest yet, except for the safety-critical software.
- What audience should be targeted as users of the formal toolkits?
  Possible choices are: academic or industrial researchers; design engineers; application domain engineers; testing engineers, or implementers. A related audience to keep in mind is tool developers. Our preference is that the toolkit should address working engineers and their daily needs.
- What level of abstraction should the toolkit address?
  For example, should the toolkit address requirements, specification, high-level design, intermediate-level design, low-level design, testing, and/or implementation. Ideally, the toolkit could be used all the way from top-level design to the implementation and testing, in a stack-like fashion, where tools communicate from one layer to another.
- What kinds of tools should be integrated in the toolkit?
  The answer to this question depends on the answers to the above questions, as well as availability of tools and possibility for collaboration with tool makers. Possible choice of tools is outlined in the previous section. We feel that the toolkit must include lightweight tools, user-friendly interface (preferably graphical), and informal tools that help automate tedious tasks.

A question remains if the tools integrated should be general-purpose or specialized. It seems the most effective to include general purpose customizable tools, so that users can tailor them to their own needs and evolve the tools as their design flow evolves.

Based on stated industrial needs, we conclude that tools must address the issue of:

- Clarity: notation must be accessible to engineers and have power of expressiveness.
- Error finding: tools must identify bugs and mistakes, rather than to prove correctness. Errors must be clearly identified, i.e. what line of code, what statement, why it is an error.
- Effectiveness: tools must be fast, easy to use, and powerful.
- Robustness: the tool itself should be as bug-free as possible; proofs, analysis, and validation should also be robust, i.e. should not easily break when specification changes.
- Extensibility: users must have the ability to customize the tool to a particular design flow.
- Automation and methodology: tools should be as automated as possible. At least, users should have some methodology to guide them.
- Reusability and change: proofs/modules should be reusable during frequent changes in requirements and/or specification.
- Scalability: tools must have some means to organize large and/or growing specifications.
- Compatibility with existing practices/tools: compatibility comes in two forms: the tool must be able to "fit" into the existing design flow; and the tool should resemble existing tools.
- Integration of graphical and textual notations, acceptable user interfaces.
- Proving application specific properties: tools must be appropriate for the task at hand.
- Tool support: the tool must have good technical support, user manual, documentation, upgrades, and training.
- Stability: how likely is it that the tool and the tool vendor will exist in the next five years?

Step 2

The main questions to be answered are:

- What kind of functionality is already existing in the current formal tools?
- Which formal tools would satisfy requirements outlined in Step 1?
- How can these tools be integrated into the design flow outlined in Step 1?

Understanding formal tools requires creating taxonomy and classification of the current formal tools, clearly outlining capabilities, inputs and outputs. This work would take some time and effort, and is recognized by the SESC Formal Methods Planning Group as a very much needed step in identifying what formal tools we have available to work with. Tool makers are requesting this information in order to be able to integrate the tools. It would be most effective to start this work with a selected subset of tools.

We also need to evaluate a selected subset of tools to identify those suitable for inclusion in the toolkit. We must examine if a tool is robust enough for industrial use. This criteria includes:

- Efficient and bug-free execution: the tool must be sufficiently tested by a variety of users, reported bugs must be fixed quickly, efficiency needs to be improved, and the improved tool distributed. It is useful to have tools and updates downloadable from the web, and to have user mailing lists and user meetings.
- Documentation: usually one of the weakest links. User manuals are frequently not clear and complete, because they are written by tool makers who understand the tool so well that they take the information for granted, or do not have the time or the preference to write reports. Error messages generated by the tools also need to be improved, to show where and what occurred. This work requires many days of "grange" work that is not publishable and not well represented in grant funding. A commercial development team could implement these user-friendly features.
- Technical support: it is necessary to have dedicated personnel, well versed in the tool, available to answer user questions. Clearly, this takes personnel time.
- Learning curve and user friendliness: the tool has to use notations that are easy to learn by the users, because industrial users do not have the time to learn complex tools on the job. Until more users are trained in formal tools, we need to resort to this measure.

Other critical issues for success of the toolkit include:

- technology transfer
- publicity and marketing
- extensibility of the tool, i.e. users must be able to customize the tool.

Understanding how the tools can be integrated might be difficult. The major barrier is that it is impossible to obtain source code for many tools. Even if source code is obtained, often collaboration with the tool maker(s) is needed. In addition, it is common that each company highly customizes off-the-shelf tools to fit into the company's design flow. If we cannot obtain tools that we could build on, we could examine source code of CAD tools constructed for government projects.

Conclusion

Industry needs assistance in producing complex, bug-free products while meeting short time-to-market deadlines. Industry is considering, or already using, formal methods. For example, many large companies have internal formal methods groups, and CAD/CASE toolkits include model checkers and equivalence checkers. Formal methods are widely accepted in hardware engineering, and less so in software engineering. In any case, formal methods have not reached a maturity stage yet and are in transition from research to practice. The consensus is that this transition can be aided by producing industrial strength formal methods tools, more technical infrastructure such as verified libraries, more (publicized) industrial case studies, and increasing the pool of trained users. It is often stated that formal methods and tools must be accessible to engineers through their notation, features, and approach.

Therefore, we must come up with a strategy that will effectively use formal tools within the currently existing parameters and work towards longer-term goals. We feel that, at this time, it seems most practical to have lightweight formal tools working in conjuction with the existing tools and practices. This approach will demonstrate usefulness of formal methods in addressing problems not easily addressable otherwise, and allow the users to get accustomed to formal methods' paradigms. More heavyweight tools would be used by a small group of experts. Meanwhile, a feedback cycle could be started: the need for better tools and positive experiences with the existing tools will cause greater industrial demand, so that academia will be willing to train more users; and vice versa. Unless there are more trained users and/or need, widespread use of formal methods will always stay at the lightweight level.

We can recommend several measures that could improve the current situation and increase industrial use of formal methods:

- Provide funding for tools that are transitioning from research environment into production environment. This would be an appropriate role for the government.
- Create opportunities for user training, either by influencing the academia or providing independent user training. An appropriate role for the government would be to provide training opportunities and thus break the chicken-and-the-egg cycle of academia waiting for a larger industrial demand, and industry demanding more when there are more trained users demanding more tools.
- Generate publicity for formal tools and research, so that the available knowledge can be disseminated through wider circles of users and different communities. For example, cross-pollinating research in hardware and software protocols would be useful to both communities. We recommend:
  - one-page fliers to be distributed at conferences
  - tool demos at a wide array of both academic and commercial conferences
  - a web page, linked to the World Wide Web Virtual Library on Formal Methods, with clear taxonomy of terms used in formal methods and classification of the tools
  - a web page, linked to the World Wide Web Virtual Library on Formal Methods page, with documentation of case studies, especially success stories. Publicized success stories could be used to justify use of formal methods.
  - a conference to present SRI International and other research of interest to the Rome Research Site to a wide community. The purpose of this conference would be to disseminate information and to provide a forum for discussion and exchange of ideas.
- Develop taxonomy of terms used in formal methods and classification of formal tools, clearly stating tools' capabilities. Ideally, we would include tools' inputs and outputs. This survey needs to be well publicized, including: a web page linked to the

World Wide Web Virtual Library on Formal Methods page; notices to e-mail lists; notices to newsgroups; and one-page fliers to be distributed at conferences. This survey would greatly benefit anyone who is searching for the right tool as well as toolkit makers. This effort would take a log time to complete fully, and, therefore, could be started for a subset of tools and/or subset of properties.

- Create verified libraries and other supporting theories, in order to build infrastructure to make formal tools easier to use. Additional research would determine the language and contents of the libraries.
- Integrate formal tools into existing tools that are already used in industry. This approach would require either access to tools' source code, or extensible tools. Industry might not be willing to disclose their design flow, and additional research needs to be done to determine what kinds of tools are needed.
- Create a framework to integrate formal tools into usable, industrial strength toolkits. Some work is already being done in this arena, mostly in Europe (please refer to the list of related projects). Integrating different tools is a difficult undertaking that might possibly requires access to the source code of various tools and collaboration with the original tool developers. The feasibility of the framework would be best illustrated by including parts of a tool into an existing and available toolkit.

Overview of Some Commonly Used Formal Tools

Toolkits Which Include Model Checkers

Model checkers are automated tools based on state space exploration. There are many model checkers in current use. They have become quite popular in the recent years, and are claimed to have "saved the reputation of formal methods" because of their ease of use and learning. Model checkers are automated tools suitable for users not trained in rigorous mathematics and theorem proving skills. Currently, model checkers are widely used in industry and in academia. Some of the most popular ones in the USA include, in alphabetical order: Murpφ, SCR, SDT, SMV, and SPIN. Model checkers popular in Europe are FDR, LOTOS, SDT, and VDM tools.

The variety in model checking tools is due to the fact that tool makers had different use of the tools in mind. Roughly speaking, we can categorize model checkers based on the language supported (which is based on application domain, level of use, and intended audience), and measures used to achieve efficiency. Priorities for choosing or creating a language could include:

- Simple notation, such as: programming-like, tabular, or graphical
- Concurrency
- Message communication
- Message buffering
- Value-passing (i.e. use of data)

- Equivalence between various specifications
- Synchronous versus asynchronous modeling

The language is also catered to the specification level it will be applied to (e.g. requirement specification). Most model checkers are intended for describing control-intensive systems and usage by "regular" engineers. Most model checkers include a symbolic simulator (it is a byproduct of storing state space), and temporal logic for describing system properties.

Measures for increasing efficiency usually involve the language in which the tool is implemented (such as C, Lisp, or ML) and algorithms for reducing state space, such as exploiting symmetry (Murpφ), hash reduction (SPIN), or automatic abstraction of unnecessary details (SCR*).

Murpφ

http://verify/stanford.edu/dill/murphi.html

Murpφ toolset is intended for asynchronous, interleaving modeling of concurrent systems, particularly hardware protocols. Murpφ toolset is based on Murpφ language, which is similar to Lisp programming language. Murpφ language has constant, type, variable and procedure declarations, rule definitions, a description of the next state, and a collection of invariants (Boolean expressions that reference a variable.) Murpφ includes a model checker. It is possible to check for invariant violations, error statements, assertion violations, and deadlock. It is possible to interface to SVC. Methods for increasing efficiency of Murpφ include: state reduction by using symmetry, execution of action rules in reverse order, and keeping track of the number of processes in a particular state instead of the processes themselves.

Symbolic Model Verifier (SMV)

ftp://emc.cs.cmu.edu/pub/

SMV was intended for hardware systems. The SMV model checker is based on language SMV, which hierarchically describes finite-state machines at any level of detail, both synchronous and asynchronous. SMV uses CTL for verification. CTL is an extension of Boolean logic with four temporal formulas: atomic formulas, propositional logic, next-state logic, and "until" logic. CMU SMV is implemented using BDDs. There is also SMV tool from CadenceBerkeley Labs.

Software Cost Reduction (SCR*) toolkit

http://www.csr.ncl.ac.uk/projects/FME/InfRes/tools/fmtdb044.html,
http://www.itd.nrl.navy.mil/ITD/5540/personnel/heitmeyer.html.

SCR* is intended for modeling safety-critical software systems at the level of requirements specification. It is meant to be used by application engineers with no training in formal methods and no access to formal methods experts. SCR* uses SCR tabular notation to describe required system behavior as the synchronous composition of non-deterministic environment and deterministic system. Each SCR specification is organized into two parts: dictionaries (which define static information such as variable names), and tables (which specify how the variables change in response to the input events.) the SCR* toolkit contains the model checker SPIN and several lightweight tools:

- Specification editor
- Graphical dependency browser: aids I understanding dependencies among variables
- Consistency checker: exposes syntax errors, variable name discrepancies, missing cases, unwanted non-determinism, and circular definition.
- Simulator

It is possible to automatically generate state invariants (i.e. properties that hold true in a given state) from the requirements specification. Before calling SPIN, it is possible to automatically reduce state space by eliminating unnecessary details.

SPIN
http://www.dcs.gla.ac.uk/~tfm/fmt/hol.html
SPIN is Bell Labs' tool intended for concurrent systems, particularly communication protocols. The underlying language is PROMELA (Protocol or Process Meta Language), which is a C-like non-deterministic language based on Hoare's CSP. SPIN contains a symbolic simulator, model checker, and verification with linear temporal logic and invariants. SPIN uses hash algorithms to reduce state space. SPIN is implemented in C, and its source code is publicly available.

SDT toolkit is described in Appendix B as a part of Telelogic Tau toolkit.

CAD/CASE formal tools will be described in Appendix B.

Theorem Provers

What we call "theorem provers" are actually mechanized proof assistants. These automated tools are powerful in their ability to specify and verify large amounts of data, and require at least six months of learning time.

A Computational Logic for Applicative Common Lisp (ACL2)
http://www.cs.uwyo.edu/~cowles/acl2
ACL2 uses first-order logic of total recursive functions. The syntax is that of Common Lisp, which allows interface to programming in Lisp. The ACL2 theorem prover can be run in completely automatic mode or with user interaction.

Prototype Verification System (PVS)
http://pvs.csl.sri.com/
PVS uses a language based on a variant of higher-order logic. It contains a theorem prover, interface to temporal model checking, and consistency and completeness check tools for tabular specifications of the kind advocated by Parnas.

HOL
http://www.dcs.gla.ac.uk/~tfm/fmt/hol.html
The HOL theorem proving system is based on higher order logic, and uses Standard ML of New Jersey (SML/NJ) as the specification language. Since SML/NJ is public, HOL is extensible.

Stanford Validity Checker (SVC)
http://agamemnon.stanford.edu/~levitt/vc/index.html
SVC is intended to be used as a fast decision procedure for validating logical expressions, and was used successfully in that capacity in PVS. A PVS proof would be reduced to a smaller subgoal, and SVC can be used on that subgoal. SVC can be used for static analysis of formal, high-level specifications of safety critical software. SVC is based on first-order logic. The Web page has little information about the other features of the tool.

Consistency Tools

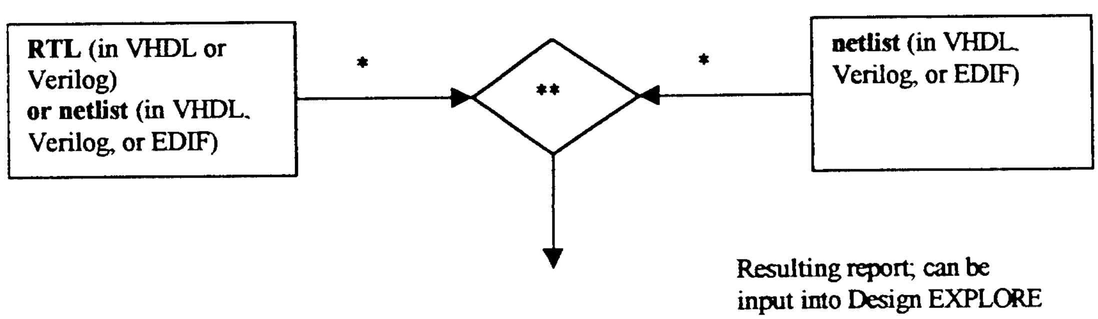
Most toolkits that include model checkers either include consistency tools or can be used for consistency checking. Theorem provers also can be used for consistency checking. Therefore, we could use SMV, PVS, SCR, HOL, and many other tools.

Formal Verification Tools in Hardware Toolkits

There are many formal hardware tools, but we will focus on several.

Design Suite of Tools, by Chrysalis Symbolic Design, Inc., http://www.chrysalis.com. Chrysalis specializes in formal tools for CAD, and its entire product line consists of three formal tools. The tools can operate independently or as a toolkit, either stand-alone or as a part of a CAD/CASE toolkit. The interoperability with regular CAD/CASE tools is achieved by using Verilog, VHDL and EDIF for input/output to Design tools.

Design VERIFYer is a formal equivalence checker. Chrysalis markets it as a replacement for gate-level regression simulation. The tool can perform hierarchical verification: RTL to RTL, RTL to gate, gate to gate or RTL to switch. The block diagram of the tool is given below.



* convert into Chrysalis logic representation
** compare the logic driving each state bit

Design EXPLORE is an interactive debugger, which takes as input RTL, netlist, or library sources in VHDL, Verilog, EDIF, or HDL and converts them into Chrysalis logic.

Design INSIGHT Assertions is a model checker which uses assertions to prove desired properties. Assertions are Boolean statements that are true at certain points in code. Assertions can be used to state properties to be proved, such as input and output module requirements, and to state assumed properties that hold true. Properties can be expressed in VHDL, Verilog, or tool command language (tcl), Temporal operators are also available. Chrysalis is marketing this model checker as a time-saving tool, because modules can be tested before the whole design is operational. Most test benches operate at the chip level, so all modules have to working in order to test the logic.

The following is taken from Chrysalis' press release:

> November 16, 1998 - Toshiba America Electronics Corporation (TAEC) has selected Chrysalis' Design VERIFYer equivalency checker for their gate-to-gate formal verification. "We were spending thousands of hours on gate-level simulation for regression analysis and needed to streamline our design process through the use of formal verification. Chrysalis' Design VERIFYer is a proven solution for gate-to-gate comparisons," said Jeff Berkman, vice president, SLI engineering for TAEC. "Additionally, Chrysalis' application expertise and support is excellent."

FormalCheck, by Lucent Technologies Bell Labs Design Automation, http://www.lucent.com/. FormalCheck is based on VHDL/Verilog. The following is an excerpt from the hardware industry news:

> "Lucent Technologies Inc.'s Bell Labs Design Automation (Murray Hill, N.J) has been awarded a contract for more than $1.25 million by Silicon Graphics/Cray Research (Mountain View, Calif.) for the follow-up purchase of additional licenses of FormalCheck, its system-level model-checking technology."

The tool has been given to Cadence Berkeley Laboratories, and we could not find current tool brochures.

Static Verifyer, by Synopsys Inc., http://www.synopsys.com/products/staticverif/staticverif.html. Synopsys is marketing Static Verifyer toolkit approach as the substitute for the current practice. Currently, engineers do not simulate much at the RT level, but simulate extensively at the gate-level. Full chip gate-level simulations run for days, up to several weeks. Synopsys' solution is to: simulate at the RT level using Synopsys' **VCS** (Verilog simulator) and **Cyclone** (VHDL simulator), and perform static verification at the gate level using Static Verifyer. Static verification consists of static timing and equivalence checking. **Prime Time** is Synopsys' static timing tool, which checks if the delays of all paths in a circuit satisfy timing assertions. Synopsys **Formality** is equivalence checker, which compares functional equivalency of the RTL source to the post-synhesis netlist. The last part of the Static Verifyer toolkit are tools used for testing: Design Compiler Expert Plus, TestGen and PathTest.

Static Verifyer requires synthesis-based design flow and synchronous design.

Eagle™, by Synopsys Inc., http://www.synopsys.com/products/hwsw/eagle_ds.html. Eagle™ is a toolkit for hardware-software co-design, intended for be used as a part of XRAY® Debugger software development toolkit from Mentor Graphic Corp.. Debugger works with Microtec C and C++ compilers. Eagle™ toolkit consists of VHDL and Verilog simulators and VERA™

Summary of Related Projects

Telelogic Tau toolkit, by Telelogic Inc., http://www.telelogic.com. This is a commercial toolkit for all phases of software development. Telelogic Inc. is a Sweden-based company with offices in New Jersey, California and Illinois. The entire toolkit costs about $10,000. The tools are: ORCA for UML-based requirement specification, SDT for system design and implementation in SDL, and ITEX for testing with TTCN. SDL is a graphical internationally standardized formal language by ITU. SDL is an object oriented language that can describe concurrent finite state machines. It accepts data inputs as abstract data type (ADT) or Abstract Syntax Notation One (ASN.1). The Telelogic Tau toolkit automatically generates executable code, and automatically generates executable test vectors. SDL and TTCN have the ability to work with ASN.1. SDT supports writing specifications of CORBA objects in Interface Description Language (IDL). An observation: several months after viewing the Telelogic Tau demo at WIFT'98, we were contacted by a sales person. We were impressed by Telelogic's commitment to sales and marketing.
A similar toolkit, called **ObjectGEODE**, based on OMT, SDL, and MSC, is produced by Verilog, USA/France.

Open Mechanized Reasoning Systems (OMRS) project, by John McCarthy, at all. http://www.mrg.dist.unige.it/omrs/index.html. This Internet site cannot be opened, and information would have to be obtained in another way.

The 15th International Conference on Automated Deduction (CADE-15), July 5-10, 1998, Lindau, Germany. Topic: Integration of Deduction Systems. http://i12www.ira.uka.de/Workshop/cfp.html.

Proof and Specification Assisted Design Environments (PROSPER) Project, http://www.dcs.gla.ac.uk/prosper. This is a recently started 3-year joint project between the Universities of Glasgow, Cambridge, Edinburgh, and Karlsruhe, and commercial tool builders Prover Technology AB (Sweden) and IFAD (Denmark). Prover Technology AB sells theorem proving technology, training and research. Prover Technology AB invented Stålmarck's algorithm, which became very popular in Europe because it provides a fast and efficient alternative to BDDs for verification purposes. Prover Technology AB implements this algorithm in its Otter theorem prover, and was able to verify formulas up to 350,000 connectives [FMCAD'98], p.82. IFAD produces the VDM-SL Toolbox, currently used at more than 50 sites, mostly in large European companies. Both companies plan to use the PROSPER toolkit to improve their commercial products. The goal of PROSPER project is to have a user-friendly, flexible, extensible toolkit that can easily integrate with various commercial tools. The PROSPER toolkit will provide a GUI interface and a common theorem proving support for a software and a hardware toolkit. The software toolkit is the VDM-SL toolbox produced by IFAD. The hardware CAD toolkit will be built by PROSPER and will be based on VHDL/Verilog. Other features of the project include: next generation interfaces and tools for requirements specification, using natural language and timing diagrams; and open proof architecture, containing an API for CAD/CASE tool integration, the core proof engine, and a plug-in interface.

Universal Formal Methods Workbench (UniForM), http://www.informatik.uni-bremen.de/~agbkb/UniForM. This is a (completed?) 3-year joint project between the University of Bremen and Carnegie Mellon and Stanford Universities, and companies SRI International and the Kestrel Institute. The goal is to perform application-driven basic research and develop reliable software for industry. The project will result in a tool kit with a common user interface, to be available in public domain in 1998. "UniForM is a specific framework instantiated with specific tools to handle communicating distributed systems and real time requirements." The project includes a case study, control of a safety critical system, and using the example of developing a decentralized central unit for a single track tramway networks. This project follows the V-model, developed by the German Ministry of Interior and accepted as standard in German software industry. V-model is a standard development process model for planning and executing information technology projects. UniForM combines duration calculi, CSP (with FDR), Z (with type checker) and PLC. The tools are integrated using development by transformation from one tool to another.

ESPRESS project, http://www.first.gmd.de/~espress. This project was sponsored by Daimler-Benz AG, and carried through cooperation between German government and universities. ESPRESS is designed for developing software for complex, safety-critical embedded systems, such as intelligent cruise control systems. The toolkit covers the entire product cycle in application areas of automobile electronics and traffic light control. ESPRESS toolkit provides one interface to many tools: Statecharts and Z notation are used to represent requirements, Isabelle/HOL-Z tool is used for validation, verification and generation of test cases, and commercial tools such as Statemate are used for editing, type checking, and specification validation. ESSPRESS was presented at the Z User meeting (ZUM), International Conference of Z Users, September 1998, Berlin, Germany. The conference was sponsored by Daimler-Benz.

Project KorSys (Korrekte Software für Sicherheitskritische Systeme, or Correct Software for Safety-critical Systems), http://www4.informatik.tu-muenchen.de/proj/korsys/all index.html, is a joint effort between companies BMW, Siemens AG, FZI, and ESG, and universities Technical University of Munich, and the University of Oldenburg. This project deals with methods and tools for reactive and finite-state systems. Application areas include avionics system at ESG and locking and hybrid systems at BMW. The project page contains little information in English.

Ptolemy Project, http://ptolemy.eecs.berkeley.edu/. This is University of California at Berkeley ongoing effort, led by Edward Lee. The project includes a model checker MOCHA by Tom Henzinger from UC Berkeley, but is mainly integrating non-formal tools. Nevertheless, the project is of interest to us because it provides expertise in integration of diverse methods and tools. The project's goal is to develop techniques ad tools for modeling heterogeneous, reactive systems, particularly embedded systems. Ptolemy can model heterogeneous systems such as those including hardware and software, analog and digital, and electrical and mechanical devices. Ptolemy can also model systems that are complex in the sense that they mix widely different operations, such as signal processing, feedback control, sequential decision making, and user interfaces. The project's web page claims that "using Ptolemy software, a high-level dataflow model of a signal processing system can be connected to a hardware simulator that in turn may be connected to a discrete-event model of a communication network." It would be worth further research to investigate how the Ptolemy Project could provide us with a wealth of experience in tool integration. Ptolemy research influenced products by Cadence, Motorola, HP, and many other companies.

Formal Methods Europe (FME), http://www.cs.tcd.ie/FME/, is a European organization supported by the Commission of the European Union, with the mission of promoting and supporting the industrial use of formal methods for computer systems development.

Forschungszentrum Informatik (the Research Center for Information Technologies, FZI), at the University of Karlsruhe, Germany, http://www.fzi.de/prost/prost_e.html. FZI is an industry-oriented research institution with the mission to implement technology transfer from the university to the industry. It is not entirely clear from the web page, but it appears that FZI is a commercial company rather than government or university sponsored center. The FZI Software Engineering Department focuses on: the development of methods and tools for the economical development and use of software; the methodology of design and constitution of reusable software; development of open programming environments; integration strategies of software tools; practical applicability of formal methods; and the reorganization and re-engineering of legacy systems.

References

[CD98] "Formal Verification: Essential for Complex Designs." *Computer Design* 37(6), June 1998.

[ClWi96] Edmund M. Clarke, Jeannette M. Wing, et. al. "Formal methods: State of the Art and Future Directions." *ACM Computing Surveys*, 28(4):626-643.

[21EC98] *21st Century Engineering Consortium Workshop: a forum on formal methods education*. March 1998, Melbourne, FL. Sponsored by Air Force

Rome Research Site, organized by Michael Nassiff. For a draft of the workshop report, contact Steven Johnson, Indiana University. Affiliated web site: http://www.cs.indiana.edu/formal-methods-education.

[Heit98] Constance L. Heitmeyer. "On the Need for 'Practical' Formal Methods," Formal Techniques in Real-Time and Real-Time Fault-Tolerant Systems, *Proceedings of the 5th Intern. Symposium* (FTRTFT'98), Lyngby, Denmark, September 14-18, 1998, LICS 1486, pp. 18-26, (invited paper).

[Kits92] D.H. Kitson and S. Masters, *An Analysis of SEI Software Process Assessment Results: 1987-1991*, Software Engineering Institute, CMU/SEI-92-TR-24, July 1992.

[NSF96] Gaetano Borriello (ed.), Future Research Directions in CAD for Electronic Systems. Workshop sponsored by NSF/CISE/MIPS, Seattle, May 1996. http://www.cise.nsf.gov/ccr/nsf-workshop/index.html

[NSF98] "Final report: NSF Workshop on Billion-Transistor Systems," Princeton, NJ, March 1998. http://www.ee.princeton.edu/~wolf/nsf-workshop/final-report.html

[Paul93] Mark C. Paulk, at all., "Key Practices of the Capability Maturity Model, Version 1.1." Software Engineering Institute, CMU/SEI-93-TR-25, February 1993. http://rbse.ics.nasa.gov/CMM/TR25/tr25_o1.html

[FMCAD98] Ganesh Gopalakrishnan, Phillip Windley (eds.), *Proceedings of Second International Conference on Formal Methods in Computer-Aided Design (FMCAD '98)*, Palo Alto, CA, USA, November 1998. Lecture Notes in Computer Science 1522, Springer.

[PROSP] "Proof and Specification Assisted Design Environments (PROSPER) Project Programme." ESPRIT LTR Project 26241, March 1998. Contact: T.F. Melham, Department of Computing Science, University of Glasgow, G128QQ, Scotland. http://www.dcs.gla.ac.uk/prosper.

[SDL92] "Specification and Description Language (SDL-92)." Z.100 (3/93). "SDL Combined with ASN.1." Z.105 (3/95). ITU General Secretariat, Sales Section, Place de Nations, CH-1211 Geneva 20.

[STTT] Springer International Journal on Software Tools for Technology Transfer (STTT). http://sttt.cs.uni-dortmund.de. Since September 1997.

[TTCN] "Framework and Methodology for Conformance Testing of Implementation of OSI and CCITT Protocols: TTCN Standard Definition." X.290, ISO/IEC-9646-3.

ROBUST SPREAD SPECTRUM COMMUNICATIONS: ADAPTIVE INTERFERENCE MITIGATION TECHNIQUES

Stella N. Batalama
Assistant Professor
Department of Electrical and Computer Engineering
E-mail: batalama@eng.buffalo.edu

The State University of New York at Buffalo
201 Bell Hall
Buffalo, NY 14260

Final Report for:
Summer Research Extension Program
Rome Laboratory

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

Rome Laboratory

and

State University of New York at Buffalo

December 1998

ROBUST SPREAD SPECTRUM COMMUNICATIONS: ADAPTIVE INTERFERENCE MITIGATION TECHNIQUES

Stella N. Batalama
Assistant Professor
Department of Electrical and Computer Engineering
State University of New York at Buffalo
E-mail: batalama@eng.buffalo.edu

Abstract

We consider the problem of adaptive detection of a Spread-Spectrum (SS) signal in the presence of unknown correlated SS interference, multipath signal reception, and additive impulsive noise. The proposed general SS receiver structure is comprised by a vector of adaptive chip-based non-linearities followed by an adaptive Auxiliary-Vector linear tap-weight filter. The non-linear receiver front-end adapts itself to the unknown prevailing noise environment providing robust performance over a wide range of underlying noise distributions while the adaptive Auxiliary-Vector linear tap-weight filter allows rapid adaptation for SS interference suppression with a limited data record. We examine a clipper-type, a puncher-type and a Hampel-type non-linearity. Numerical and simulation results demonstrate the performance of the proposed approach and offer comparisons with the conventional Minimum-Variance-Distortionless-Response (MVDR) filter, as well as the MVDR filter preceded by a vector of adaptive chip-based non-linearities. The latter structure is a generalization for the multipath case of the scheme originally proposed in [8], [9].

ROBUST SPREAD SPECTRUM COMMUNICATIONS:

ADAPTIVE INTERFERENCE MITIGATION TECHNIQUES

Stella N. Batalama

I. Introduction

Signal detection in the presence of impulsive channel noise has been considered in [1]-[5], while direct-sequence spread-spectrum (DS-SS) signal detection under similar channel models has been studied in [6]-[9]. Receiver proposals in [6],[7] involve the use of either a conventional matched filter or a majority-vote receiver (hard-limiter non-linearity per chip followed by signature matched-filtering). In [6] it is reported that neither one of the above proposals is universally effective against the combination of SS interference and non-Gaussian impulsive noise. In [9], adaptive receivers are proposed that are comprised by a vector of adaptive chip-based non-linearities followed by an adaptive linear tap-weight filter. The structures proposed in[8]-[9] tap the relative merits of both non-linear and linear signal processing and exhibit superior BER performance in the presence of combined impulsive and SS interference. In particular, the non-linear receiver front-end adapts itself to the unknown prevailing noise environment providing robust performance over a wide range of underlying noise distributions, while the adaptive linear tap-weight filter that follows the non-linearly processed chip-samples combats effectively the SS interference. This paper extends the work in [8]-[9] in the following aspects. It shifts the receiver design objective to superior bit error rate (BER) performance under rapid, short-data-record adaptation and the system model is generalized to account for multipath signal reception. An additional non-linear Hampel-type pre-processor is considered that combines the characteristics of the non-linearities studied in [8]-[9].

II. System Model

Our system model involves a signal of interest with code $\mathbf{S}_0$ of length L chips (if T is the symbol period and T_c is the chip period then $L = T/T_c$), M multipaths, (K-1) multipathed DS-SS interferers with signatures $\mathbf{S}_k$, $k = 1, \ldots, K-1$, and non-Gaussian (impulsive) interference. For notational simplicity and without loss of generality we choose a synchronous set-up. We assume that the multipath spread is of the order of a few chip intervals and since the signal is bandlimited to $B = 1/T_c$, the tap-delay line channel model has taps spaced at chip intervals T_c. The chip-synchronous received signal vector during a symbol interval that results after conventional chip-matched filtering and sampling at the chip rate comprises of $L + M - 1$ samples and is given by

$$\mathbf{r} = \sum_{k=0}^{K-1} b_k \sum_{m=1}^{M} c_{k,m} \sqrt{E_k} \mathbf{S}_{k,m} + \mathbf{n}, \tag{1}$$

where, with respect to the k-th SS signal, b_k and E_k are the transmitted bit and energy, respectively, and $\{c_{k,m}\}$ are the coefficients of the frequency-selective slowly fading channel modeled as independent zero-mean complex Gaussian random variables that remain constant within a bit interval. $\mathbf{S}_{k,m}$ is the signature of the k-th SS signal corresponding to its m-th path (we assume 0-padding for the expansion of $\mathbf{S}_k$ to $\mathbf{S}_{k,m}$) and $\mathbf{n}$ is assumed to be complex non-Gaussian impulsive noise.

An equivalent representation of the received signal $\mathbf{r}$ is given by

$$\mathbf{r} = \sqrt{E_0} b_0 \mathbf{w}^{(0)}_{\text{R-MF}} + \mathbf{I} + \mathbf{n}, \tag{2}$$

where $\mathbf{w}^{(0)}_{\text{R-MF}} = \sum_{m=1}^{M} c_{0,m}\mathbf{S}_{0,m}$ is the effective channel processed signature of the SS signal of interest (signal-*0*). In (2), **I** identifies comprehensively both the Inter-Symbol and the SS interference. We use the subscript R-MF in the effective signature notation to make an association with the RAKE Matched-Filter receiver that is known to correlate the signature $\mathbf{S}_0$ with M size-L shifted windows of the received signal (that correspond to the M paths of the channel) appropriately weighted by the conjugated channel coefficients, $c_{0,m}, m = 1, \ldots . M$.

III. Distribution-free non-linear pre-processors.

In this section we generalize the distribution-free non-linearities considered in [8]-[9] for the multipath model adopted and we also propose to study an additional Hampel-type non-linearity. The non-linearities used in the present work are defined below:

Clipper-type non-linearity

$$g_1(x) \triangleq \begin{cases} x, & \text{if } |x| \leq c, c > 0 \\ c\frac{x}{|x|}, & \text{if } |x| > c, c > 0. \end{cases} \tag{3}$$

Puncher-type non-linearity

$$g_2(x) \triangleq \begin{cases} x, & \text{if } |x| \leq c, c > 0 \\ 0, & \text{if } |x| > c, c > 0. \end{cases} \tag{4}$$

Hampel-type non-linearity

$$g_3(x) \triangleq \begin{cases} x, & \text{if } |x| < a, a > 0 \\ a\frac{x}{|x|}, & \text{if } a \leq |x| < b, a, b > 0 \\ \frac{c-|x|}{c-b}a\frac{x}{|x|}, & \text{if } b \leq |x| \leq c, a, b, c > 0 \\ 0, & \text{otherwise .} \end{cases} \tag{5}$$

In all the above non-linearities, x is a complex number and $|x|$ denotes the norm of x. The linear region of the non-linearities has the effect of passing the observations undistorted. The non-linear regions either completely reject (remove) or "correct" the observation. The latter is considered as an adjustment of the norm while maintaining the phase. The parameters a, b, and c that appear in the above non-linearities are positive cutoff parameters to be determined adaptively.

IV. MS-type linear post-processing

1. Non-multipath case

To combine the best features of linear filtering and non-linear signal processing, we consider an adaptive non-linear receiver (Figure 1) for the detection of the information bits of a spread-spectrum signal of interest in a non-Gaussian impulsive noise environment.

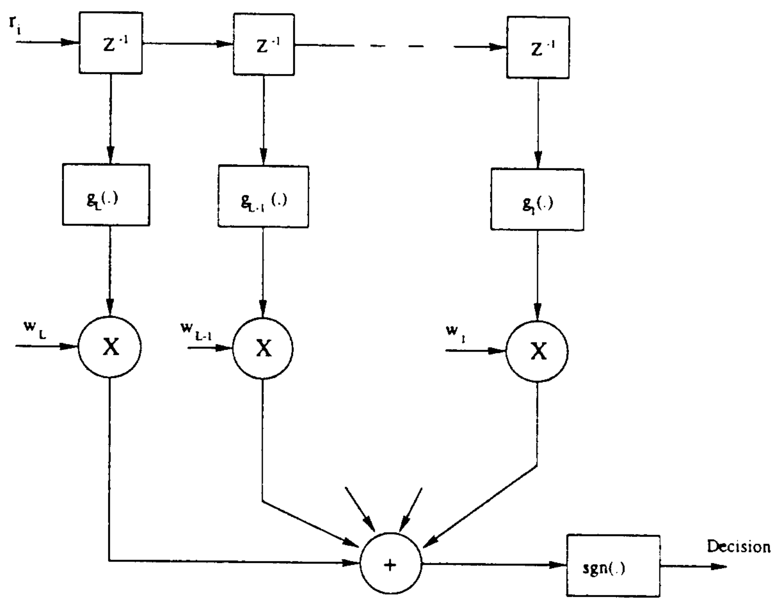


Figure 1: General Structure of a Non-Linear DS-CDMA Receiver

The receiver involves a non-linear pre-processing front-end followed by a linear tap-weight filter. The non-linearities are in general non-identical. As discussed [8]-[9], approximately optimum performance can be obtained if the non-linearities are chosen on the basis of explicit apriori knowledge of the underlying combined-noise statistics. In realistic applications, however, this information is not usually available and our approach proceeds by choosing a parametrically described distribution free non-linearity which adapts its parameters to the unknown prevailing noise environment. This way we develop a receiver that can be efficiently optimized and exhibits robust performance over a wide range of underlying noise distributions. In the following, the non-multipath case is presented in the context of either the clipper or the puncher. These non-linearities are adaptive with respect to the cutoff parameter as shown in Figure 2.

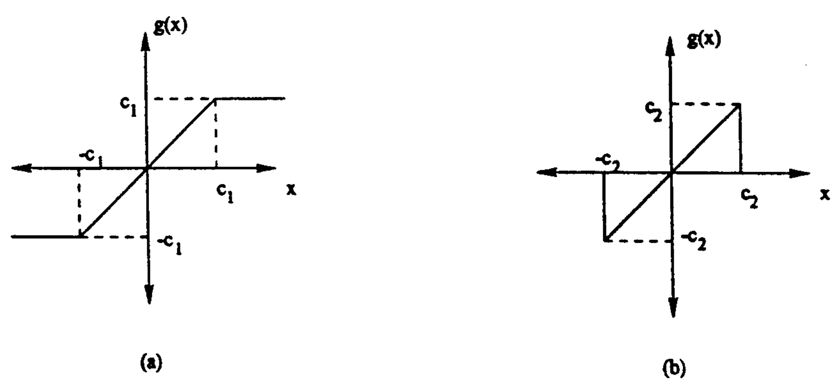


Figure 2: (a) Clipper, (b) Puncher.

The overall system parameters are determined by either a joint MSE-BER optimization algorithm, or a joint BER algotithm or a joint MSE algorithm, as described below.

A. Joint MSE-BER Optimization

In this approach the non-linearly processed chip samples drive a linear tap-weight filter whose taps are determined according to the MMSE or MVDR criterion. Under both criteria the tap-weight filters are determined by the statistics of the non-linearly processed received signal samples. Let $\mathbf{g}(\cdot) = [g_1(\cdot), \ldots, g_L(\cdot)]$ denote the vector of the non-linearities and $\mathbf{w_g} = [w_1, \ldots, w_L]^T$ denote the linear tap-weight filter parameterized by the non-linearity $\mathbf{g}(\cdot)$. Then the test statistic for the detection of the information bit of the spread-spectrum sign al of interest is determined by the inner product

$$< \mathbf{g}(\mathbf{r}), \mathbf{w_g} > \tag{6}$$

and appropriate tap-weight vectors $\mathbf{w_g}$ are given by the following proposition.

Proposition 1 *Given the non-linearity* $\mathbf{g}(\cdot)$ *then:*

(i) the MMSE linear tap-weight vector $\mathbf{w_g}$ *is given by*

$$\mathbf{w}_{\mathbf{g},MMSE} = \mathbf{R_g}^{-1} E\{\mathbf{g}(\mathbf{r}) b_0\}\,. \tag{7}$$

(ii) the MVDR linear tap-weight vector $\mathbf{w_g}$ *that is distortionless in the* $\mathbf{S}_0$ *direction is given by*

$$\mathbf{w}_{\mathbf{g},MVDR} = \frac{1}{\mathbf{S_0}^T \mathbf{R_g}^{-1} \mathbf{S}_0} \mathbf{R_g}^{-1} \mathbf{S}_0\,. \tag{8}$$

For both (i) and (ii) the covariance matrix $\mathbf{R}$ *is given by*

$$\mathbf{R_g} = E\{\mathbf{g}(\mathbf{r}) \mathbf{g}^T(\mathbf{r})\}\,. \tag{9}$$

Proof

(i) The filter $\mathbf{w}$ in (7) is the Wiener solution for a linear tap-weight filter with input vector $\mathbf{g}(\mathbf{r})$.

(ii) The filter $\mathbf{w}$ in (8) is the solution to the optimization problem that minimizes the output variance under the constraint that $\mathbf{w}^T \mathbf{S}_0 = 1$. □

Corollary 1 *If the non-linearity* $g_i(\cdot)$ *is the clipper or the puncher with cutoff parameter equal to c, then:*

(i) the MMSE linear tap-weight vector is given by

$$\mathbf{w}_{c,MMSE} = \beta_1 \mathbf{R}_c^{-1} \mathbf{S}_0 \,, \tag{10}$$

(ii) the MVDR linear tap-weight vector is given by

$$\mathbf{w}_{c,MVDR} = \frac{\beta_1}{\mathbf{S}_0{}^T \mathbf{R}_c^{-1} \mathbf{S}_0} \mathbf{R}_c^{-1} \mathbf{S}_0 \,. \tag{11}$$

For both (i) and (ii) $\mathbf{R}_c$ *is given by (32) with* $\mathbf{g}(\cdot)$ *being the vector of non-linearities denoted as* $\mathbf{g}_c(\cdot)$ *and* β_1 *is equal to the probability that* $-c \leq r_i \leq c$.

Proof

Let β_1, β_2, and β_3 denote the probabilities that $r_i \in [-c, c]$, $r_i > c$, and $r_i < -c$, respectively. If $g_i(\cdot)$ is the clipper then $E\{g_i(r_i)b_0\} = \beta_1 E\{r_i b_0\} + \beta_2 E\{cb_0\} + \beta_3 E\{-cb_0\} = \beta_1 S_{0i}$. If $g_i(\cdot)$ is the puncher then $E\{g_i(r_i)b_0\} = \beta_1 E\{r_i b_0\}$. Part (ii) can be proven similarly. □

The filters given in (i) and (ii) of Corollary 1 result in equivalent receiver structures since the decision operation $sgn(< \mathbf{g}_c(\mathbf{r}), \mathbf{w}_c >)$ is positive-scalar inv riant.

The simplicity of the particular parametric non-linearities (clipper and puncher) together with the adaptive evaluation of the cutoff parameter result in a simple structure that exhibits the favorable characteristics of both conventional linear filtering and non-linear signal processing. The cutoff parameter c is related to our confidence on the interval $[-c, c]$ and the fraction of the received data that we expect to fall in the linear region of the receiver non-linearity. Thus, in highly impulsive environments a reasonably small value of the cutoff parameter c will prohibit the contamination of the decision statistic (i.e. the output of the linear filter post-processor) by high amplitude disturbances. On the other hand, the more Gaussian (non-impulsive) the environment is, the larger the cut-off parameter value can be and the closer to the conventional linear MMSE or MVDR filter the proposed receiver structure is. In other words, the cutoff parameter c tracks the impulsiveness of the environment and the overall structure converges, as $c \to \infty$, to the conventional linear receiver. For the evaluation of the parameter c let us assume without loss of generality that the receiver employs the same non-linearity per chip sample $g_c(\cdot)$ parameterized by the cutoff parameter c. Let us then denote the output of the receiver by $u(\mathbf{r}, c) \triangleq sgn < \mathbf{g}_c(\mathbf{r}), \mathbf{w}_c >$. If we define $\rho(\cdot)$ such that

$$\rho(\mathbf{r}_0, \mathbf{r}_1; c) \triangleq \frac{1}{2}\{\pi_0[1 + u(\mathbf{r}_0, c)] + \pi_1[1 - u(\mathbf{r}_1, c)]\}, \tag{12}$$

where $\pi_0 = \pi_1 = 1/2$ are the a priori probabilities of hypothesis H_0 and H_1, then we observe that the probability of error is equal to

$$P_e(c) = E\{\rho(\mathbf{r}_0, \mathbf{r}_1; c)\}, \tag{13}$$

where $\mathbf{r}_0$ and $\mathbf{r}_1$ correspond to received data vectors from H_0 ($b_0 = -1$) and H_1 ($b_1 = +1$) respectively. Thus, the so defined function $\rho(\cdot)$ (cf. (12)) provides a measure for the distortion at the output of the receiver. Indeed, the bracket terms are both zero when the receiver makes the correct decision and strictly positive otherwise. Based on stochastic gradient techniques and exploiting the property in (13), we develop an adaptive procedure that iteratively adjusts the cutoff parameter c to minimize the probability of error at the output of the receiver:

$$c_{n+1} = c_n - \alpha_n x_n(c_n), \tag{14}$$

where

$$x_n(c) \triangleq \frac{1}{2d_n}[\rho(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; c + d_n) - \rho(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; c - d_n)], \tag{15}$$

and $\{d_n\}$ and $\{\alpha_n\}$ are monotonically decreasing sequences of positive numbers such that $\sum \alpha_n = \infty$, $\sum \alpha_n d_n < \infty$, and $\sum \alpha_n^2 d_n^{-2} < \infty$.

Summarizing, given the non-linearity $g_c(\cdot)$ and the corresponding cutoff parameter c, conditional MS optimization results in the ideal filter given by Proposition 1 and/or Corollary 1. Since the filter $\mathbf{w}$ itself is a function of c, further optimization with respect to c according to (14) isolates the filter that minimizes the probability of error given by (13). We may view the overall optimization procedure described above as a set of coupled equations comprised by eq. (14) and (8). To implement this set of coupled equations we develop and examine three alternative schemes which we call MSE-BER Type I, Type II, and Type III, respectively.

a) MSE-BER Type I

The $(n+1)$ iteration step of the algorithm is given below:

$$c_{n+1} = c_n - \alpha_n x_n(c_n)\,; \tag{16}$$

$$\mathbf{w}_{c_{n+1}} = \mathbf{R}^{-1}_{c_{n+1}} E\{\mathbf{g}_{c_{n+1}}(\mathbf{r}) b_0\}\,, \tag{17}$$

where $\mathbf{R}_{c_{n+1}} = E\{\mathbf{g}_{c_{n+1}}(\mathbf{r})\mathbf{g}^T_{c_{n+1}}(\mathbf{r})\}$. Each iteration step of the algorithm requires one sample (per hypothesis) for the cutoff parameter recursion. On the other hand, the quantities $\mathbf{R}^{-1}_{c_{n+1}}$ and $E\{\cdot\}$ for the evaluation of $\mathbf{w}_{c_{n+1}}$ are estimated by SMI (Sample Matrix Inversion) and sample averaging, respectively, *based on the whole data record.*

b) MSE-BER Type II

The $(n+1)$ iteration step of the algorithm is still given by (16)-(17) as in Type I above. However, in this case, at each iteration step of the algorithm a data block of increasing size is used. In particular, the first sample (per hypothesis) is used for the recursion of the cutoff parameter while the rest are used in the evaluation of $\mathbf{w}_{c_{n+1}}$ for SMI and sample averaging estimation of $\mathbf{R}_{c_{n+1}}$ and $E\{\cdot\}$, respectively. In comparison with Type I, Type II implementation of the joint MSE-BER algorithm utilizes fewer data for the estimation of $\mathbf{w}_{c_{n+1}}$ per iteration step. This implies that Type II is less computational intensive but is slower and performs worse than Type

I in the early steps of the algorithm.

c) MSE-BER Type III

To improve the initial BER of the MSE-BER Type II coupled recursions we develop the Type III implementation that proceeds according to the following recursions:

$$c_{n+1} = c_n - \alpha_n x_n(c_n)\,; \tag{18}$$

$$\hat{\mathbf{w}}_{c_{n+1}} = LMS(\text{initial value } \hat{\mathbf{w}}^{(0)}_{c_{n+1}}, \text{data block size } \lambda_{n+1}, \text{LMS step size}, c_{n+1}) \tag{19}$$

where $\{\lambda_n\}$ denotes the sequence of increasing data block sizes and LMS$(\cdot,\cdot,\cdot,\cdot)$ denotes an LMS loop for the evaluation of the filter in the (n+1)-iteration step of the algorithm using a data block of size λ_{n+1}. The final output of the LMS loop is $\hat{\mathbf{w}}^{(\lambda_{n+1})}_{c_{n+1}}$ where the superscript denotes the internal LMS iteration index. The initial value of the filter in the LMS loop is chosen to be equal to $\hat{\mathbf{w}}^{(\lambda_n)}_{c_n}$, i.e. $\hat{\mathbf{w}}^{(0)}_{c_{n+1}} \triangleq \hat{\mathbf{w}}^{(\lambda_n)}_{c_n}$. We also indicate the cutoff parameter c_{n+1} as well as the LMS step size as additional parameters of the LMS loop.

The receivers developed so far use different criteria for the optimization of the cutoff parameter and the linear filter (minimum BER and MMSE respectively). Below we propose, develop, and study two alternative optimization approaches that make use of the *same* criterion for joint optimization of the overall system parameters.

B. Joint BER Optimization

Let $\mathbf{a}$ denote the combined vector of the adjustable receiver parameters, i.e. the filter tap-weights and the cutoff parameter c of the non-linearity (clipper or puncher). That is,

$$\mathbf{a}^T \triangleq [a_1, a_2, \ldots, a_{L+1}] = [\mathbf{w}^T, c] = [w_1, w_2, \ldots w_L, c]\,, \tag{20}$$

where $\mathbf{w}$ is the filter tap-weight vector. Then the operation performed by the non-linear receiver on the received vector $\mathbf{r}$ parameterized by the combined vector $\mathbf{a}$ can be defined as

$$\mathcal{U}(\mathbf{r}, \mathbf{a}) \triangleq sgn < g_{a_{L+1}}(\mathbf{r}), \mathbf{w} > . \tag{21}$$

(i) Clipper Non-Linearity

We define $\zeta_{clip}(\cdot)$ such that

$$\zeta_{clip}(\mathbf{r_0}, \mathbf{r_1}; \mathbf{a}) = \pi_0 f_{c0}[\mathcal{U}(\mathbf{r}_0, \mathbf{a})] + \pi_1 f_{c1}[\mathcal{U}(\mathbf{r}_1, \mathbf{a})] \tag{22}$$

where π_0 and π_1 are the a priori probabilities of H_0 and H_1 respectively, while $f_{c0}(x) \triangleq \frac{1}{2}(1+x)$ and $f_{c1}(\cdot) \triangleq \frac{1}{2}(1-x)$. Then the probability of error is equal to

$$P_e(\mathbf{a}) = E\{\zeta_{clip}(\mathbf{r_0}, \mathbf{r_1}; \mathbf{a})\} \tag{23}$$

and the vector $\mathbf{a}$ that minimizes (23) can be found using the recursion

$$\mathbf{a}_{n+1} \; = \; \mathbf{a}_n \; - \; \alpha_n \mathbf{y}_n(\mathbf{a}_n), \tag{24}$$

where

$$\mathbf{y}_n \; = \; [\mathbf{y}_{n,1}(\mathbf{a}), \ldots, \mathbf{y}_{n,L+1}(\mathbf{a})] \quad and \tag{25}$$

$$\mathbf{y}_{n,j}(\mathbf{a}) \triangleq \frac{1}{2d_n}[\zeta_{clip}(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; \mathbf{a} + d_n \mathbf{e}_j) - \zeta_{clip}(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; \mathbf{a} - d_n \mathbf{e}_j)] . \tag{26}$$

In the above expressions, $\mathbf{e}_j$ is the $j-th$ coordinate unit vect or while $\{d_n\}$ and $\{\alpha_n\}$ are monotonically decreasing sequences of positive numbers such that $\sum \alpha_n = \infty$, $\sum \alpha_n d_n < \infty$, and $\sum \alpha_n^2 d_n^{-2} < \infty$.

(ii) Puncher Non-Linearity

In this case, the possible receiver outputs are 0, +1, −1. If $\mathcal{U}(\mathbf{r}, \mathbf{a})$ is 0, then we decide H_1 w.p. π_1 and H_0 w.p. π_0. Thus, under either hypothesis the probability of error is

$$P(e|H_0) \;\; = \;\; P[\mathcal{U}(\mathbf{r}_0, \mathbf{a}) = 1] + \pi_1 P[\mathcal{U}(\mathbf{r}_0, \mathbf{a}) = 0] \;\; \text{and}$$

$$P(e|H_1) \;=\; P[\mathcal{U}(\mathbf{r}_1, \mathbf{a}) = -1] + \pi_0 P[\mathcal{U}(\mathbf{r}_1, \mathbf{a}) = 0]. \tag{27}$$

If we define the functions $f_{p0}(\cdot)$ and $f_{p1}(\cdot)$ as

$$f_{p0}(x) = \begin{cases} 0, & x = -1 \\ \pi_1, & x = 0 \\ 1, & x = +1 \end{cases} \quad \text{and} \quad f_{p1}(x) = \begin{cases} 1, & x = -1 \\ \pi_0, & x = 0 \\ 0, & x = +1 \end{cases} \tag{28}$$

$$\text{and } \zeta_{punch}(\mathbf{r_0}, \mathbf{r_1}; \mathbf{a}) = \pi_0 f_{p0}(\mathcal{U}(\mathbf{r}_0, \mathbf{a})) + \pi_1 f_{p1}(\mathcal{U}(\mathbf{r}_1, \mathbf{a}))\,, \tag{29}$$

then

$$\begin{aligned} E\{\zeta_{punch}(\mathbf{r_0}, \mathbf{r_1}; \mathbf{a})\} \;&=\; \pi_0\{0 \cdot P[\mathcal{U}(\mathbf{r}_0, \mathbf{a}) = -1] + 1 \cdot P[\mathcal{U}(\mathbf{r}_0, \mathbf{a}) = 1] + \pi_1 \cdot P[\mathcal{U}(\mathbf{r}_0, \mathbf{a}) = 0]\} + \\ &\quad\; \pi_1\{0 \cdot P[\mathcal{U}(\mathbf{r}_1, \mathbf{a}) = 1] + 1 \cdot P[\mathcal{U}(\mathbf{r}_1, \mathbf{a}) = -1] + \pi_0 \cdot P[\mathcal{U}(\mathbf{r}_1, \mathbf{a}) = 0]\} \\ &=\; \pi_0 P(e|H_0) + \pi_1 P(e|H_1) = P(e). \end{aligned} \tag{30}$$

and the same recursion as in (24) with $\zeta_{punch}(\cdot)$ in place of $\zeta_{clip}(\cdot)$ yields the appropriate vector $\mathbf{a}$. We note that when $\pi_1 = \pi_0 = \frac{1}{2}$ then $f_{c0}(\cdot) = f_{p0}(\cdot)$ and $f_{c1}(\cdot) = f_{p1}(\cdot)$, which implies that $\zeta_{clip}(\cdot) = \zeta_{punch}(\cdot)$.

C. Joint MSE Optimization

(i) Clipper Non-Linearity

We recall that $\mathbf{a}^T = [\mathbf{w}^T, c] = [w_1, \ldots, w_{L+1}, c]$. Then the MSE between the output of the linear filter and a desired bit sequence of the spread-spectrum signal of interest is given by

$$J \;=\; E\{(<\mathbf{w}, g_c(\mathbf{r})> - b_0)^2\} \;=\; E\{(\sum_{i=1}^{L} w_i g_c(r_i) - b_0)^2\} \tag{31}$$

and the MS-adaptive implementation for the combined parameter vector $\mathbf{a}$ is given by the following recursion:

$$\mathbf{a}(n+1) = \mathbf{a}(n) - \mu \hat{\nabla}(J)(n), \tag{32}$$

where μ is the constant step-size parameter and $\hat{\nabla}$ denotes the instantaneous estimate of the gradient; that is

$$\hat{\nabla}(J) = [\hat{\nabla}_1(J), \ldots, \hat{\nabla}_L(J), \hat{\nabla}_{L+1}(J)]. \tag{33}$$

$$\hat{\nabla}_i(J) = \frac{\hat{\partial}}{\partial w_i}(J) = 2(\sum w_i g_c(r_i) - b_0) g_c(r_i), \quad i = 1, \ldots L, \tag{34}$$

$$\hat{\nabla}_{L+1}(J) = \frac{\hat{\partial}}{\partial c}(J) = 2(\sum w_i g_c(r_i) - b_0) \sum w_i \frac{\partial}{\partial c} g_c(r_i). \tag{35}$$

Also, since $g_c(\cdot)$ is the clipper, we can write:

$$g_c(r_i) = r_i[\eta(r_i + c) - \eta(r_i - c)] + c\eta(r_i - c) - c\eta(-r_i - c), \tag{36}$$

$$\frac{\partial}{\partial c} g_c(r_i) = \eta(r_i - c) - \eta(-r_i - c), \tag{37}$$

where $\eta(\cdot)$ is the unit-step function, and c is a positive scalar.

(ii) Puncher Non-Linearity

Since the partial derivative with respect to the cutoff parameter c of the MSE at the output of a system that involves the puncher non-linearity is zero almost everywhere, joint MSE optimization is not feasible in this case.

2. Multipath case

A solution for the suppression of SS-interference, intersymbol interference and non-Gaussian (impulsive) noise that assumes the least information about these signal disturbances in the presense of multipath (i.e. only the effective code of the signal of interest is assumed known) can be obtained by an adaptive linear filter designed according to a Minimum Variance Distortionless Response (MVDR) criterion preceded by a vector of adaptive chip-based non-linearities. This

MVDR-type receiver is a generalization of the receiver in [8]-[9] for the multipath case. The linear post-processor is a RAKE-type filter that minimizes the output variance subject to the constraint that the filter remains distortionless in the normalized "effective" direction/signature $\mathbf{w}^{(0)}_{\|\text{R-MF}\|}$ of the user of interest. It is given by

$$\mathbf{w}^{(0)}_{\text{g,R-MVDR}} \triangleq \frac{\mathbf{R_g}^{-1}\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}{\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\mathbf{R_g}^{-1}\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}, \tag{38}$$

where $\mathbf{R_g}$ denotes the autocorrelation matrix of the size $(L+M-1)$ non-linearly processed received signal that includes all M resolvable paths (L is the signature length), i.e. $\mathbf{R_g} = E\{\mathbf{g}(\mathbf{r})\mathbf{g}^H(\mathbf{r})\}$. To form, however, the MS-optimum adaptive filter $\mathbf{w}^{(0)}_{\text{g,R-MVDR}}$, exact knowledge of the size $(L+M-1)$ data autocorrelation matrix $\mathbf{R_g}$ is required. Since $\mathbf{R_g}$ is unknown we may use available non-linearly processed received data to form a sample-average estimate $\hat{\mathbf{R}}_\mathbf{g} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{g}(\mathbf{r}_n)\mathbf{g}^H(\mathbf{r}_n)$ and then proceed with Sample-Matrix-Inversion (SMI). In this case

$$\hat{\mathbf{w}}^{(0)}_{\text{g,R-MVDR}} = \frac{\hat{\mathbf{R}}_\mathbf{g}^{-1}\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}{\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\hat{\mathbf{R}}_\mathbf{g}^{-1}\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}. \tag{39}$$

In the above expressions, the normalized effective signature is given by

$$\mathbf{w}^{(0)}_{\|\text{R-MF}\|} = \frac{\mathbf{w}^{(0)}_{\text{R-MF}}}{\|\mathbf{w}^{(0)}_{\text{R-MF}}\|}. \tag{40}$$

Finally, the decisions are made according to

$$\hat{b}_0 = sgn(Re[\hat{\mathbf{w}}^{(0)H}_{\text{g,R-MVDR}}\mathbf{g}(\mathbf{r})]), \tag{41}$$

where sgn$(\cdot)$ denotes the sign operation.

V. Proposed algorithmic developments

It is known that batch SMI adaptive procedures outperform recursive constraint-LMS adaptive implementations of the MVDR solution in terms of small-sample convergence characteristics.

Apart from the computational complexity of the matrix inversion operation, $N \geq L + M - 1$ non-linearly processed data samples are required for the estimated covariance matrix $\hat{\mathbf{R}}_{\mathbf{g}} \triangleq \frac{1}{N}\sum_{n=1}^{N}\mathbf{g}(\mathbf{r})\mathbf{g}(\mathbf{r})^H$ to be invertible with probability 1 and in fact data sizes many times the filter length $(L + M - 1)$ are necessary for $\hat{\mathbf{w}}^{(0)}_{\mathbf{g},\text{R-MVDR}} = K\hat{\mathbf{R}}_{\mathbf{g}}^{-1}\mathbf{w}^{(0)}_{\text{R-MF}}$, $K = [\mathbf{w}^{(0)H}_{\text{R-MF}}\hat{\mathbf{R}}_{\mathbf{g}}^{-1}\mathbf{w}^{(0)}_{\text{R-MF}}]^{-1}$, to approach reasonably well the performance characteristics of the "ideal" $\mathbf{w}^{(0)}_{\mathbf{g},\text{R-MVDR}}$ filter, where "ideal" refers to the use of an asymptotically large data record (perfectly known $\mathbf{R}_{\mathbf{g}}$). Unfortunately, for typical L and M values and data transmission rates, filter adaptation over that many symbol intervals may not keep up with typical channel fluctuation (fading) rates. This discussion brings us to the core concept of this section. That is to develop an SS-receiver that (i) operates in the presence of both SS and impulsive interference, (ii) taps the merits of both non-linear signal processing and adaptive linear filtering, (iii) exhibits inherently low computational complexity and, at the same time, (iv) maintains superior short-data-record filtering performance.

To gain some insight with respect to the proposed algorithmic framework, we consider a canonical representation of the linear filter post-processor as depicted in Figure 3.

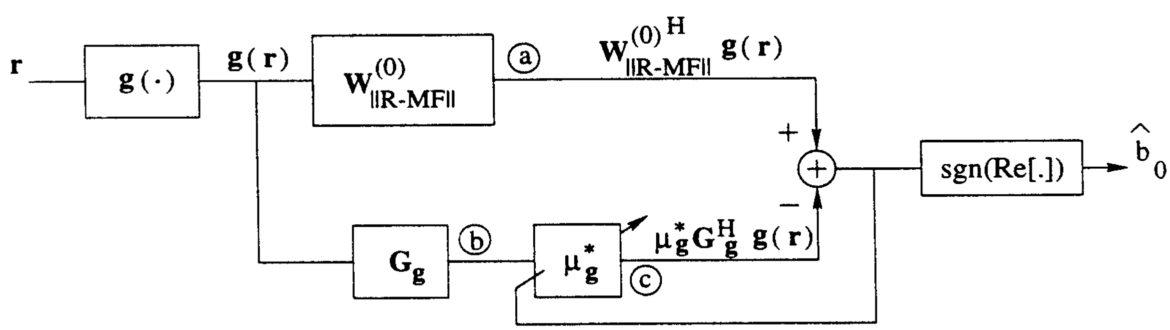


Figure 3: Proposed receiver structure.

In this figure we look at the class of scalar parameterized FIR linear post-processing filters of the form

$$\mathbf{w}^{(0)}_{\text{PROP}} \stackrel{\triangle}{=} \mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \mu_{\mathbf{g}} \mathbf{G}_{\mathbf{g}}, \tag{42}$$

where $\mu_{\mathbf{g}}$ is an arbitrary complex-valued parameter, and $\mathbf{G}_{\mathbf{g}}$ is an arbitrary "auxiliary" complex vector of dimension $L + M - 1$ that is orthonormal with respect to $\mathbf{w}^{(0)}_{\|\text{R-MF}\|} \stackrel{\triangle}{=} \mathbf{w}^{(0)}_{\text{R-MF}} / \|\mathbf{w}^{(0)}_{\text{R-MF}}\|$:

$$\mathbf{G}^{H}_{\mathbf{g}} \mathbf{w}^{(0)}_{\|\text{R-MF}\|} = 0 \quad \text{and} \quad \|\mathbf{G}_{\mathbf{g}}\| = 1. \tag{43}$$

Both $\mu_{\mathbf{g}}$ and $\mathbf{G}_{\mathbf{g}}$ are to be designed. Bit detection is made as follows:

$$\hat{b}_0 = sgn(Re[\hat{\mathbf{w}}^{(0)H}_{\text{PROP}} \mathbf{g}(\mathbf{r})]). \tag{44}$$

We note that both $\mathbf{G}_{\mathbf{g}}$ and $\mu_{\mathbf{g}}$ are parametrized by the $L \times 1$ vector of non-linearities $\mathbf{g}(\cdot)$ and for any $\mathbf{G}_{\mathbf{g}}$ satisfying the constraints in (43), the filter $\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \mu_{\mathbf{g}} \mathbf{G}_{\mathbf{g}}$ is already "distortionless" in the $\mathbf{w}^{(0)}_{\|\text{R-MF}\|}$ direction of interest since $\mathbf{w}^{(0)H}_{\text{PROP}} \mathbf{w}^{(0)}_{\|\text{R-MF}\|} = 1$. For an arbitrary but fixed auxiliary vector $\mathbf{G}_{\mathbf{g}}$, the filter in (42) can be viewed as scalar parameterized and optimized with respect to the complex scalar $\mu_{\mathbf{g}}$ only. The MS-optimum value of $\mu_{\mathbf{g}}$ is the value that minimizes the MSE $E\{|\mathbf{w}^{(0)H}_{\|\text{R-MF}\|} \mathbf{g}(\mathbf{r}) - \mu^{*}_{\mathbf{g}} \mathbf{G}^{H}_{\mathbf{g}} \mathbf{g}(\mathbf{r})|^2\}$ between points (a) and (c) in the block diagram filter representation in Figure 3.

The solution is simply

$$\mu_{\mathbf{g}} = \frac{\mathbf{G}_{\mathbf{g}}{}^{H} \mathbf{R}_{\mathbf{g}} \mathbf{w}^{(0)}_{\|\text{R-MF}\|}}{\mathbf{G}_{\mathbf{g}}{}^{H} \mathbf{R}_{\mathbf{g}} \mathbf{G}_{\mathbf{g}}}, \tag{45}$$

where $\mathbf{R}_{\mathbf{g}} = E\{\mathbf{g}(\mathbf{r})\mathbf{g}(\mathbf{r})^{H}\}$.

Now, we return to the problem of selecting an auxiliary vector subject to an appropriately chosen criterion. The selection criterion for the auxiliary vector $\mathbf{G}_{\mathbf{g}}$ that we propose to study in this paper is motivated by the MSE interpretation of the filter in Figure 3 and is the maximization of the magnitude of the cross-correlation between points (a) and (b) subject to the constraints in (43):

$$\mathbf{G_g} = \arg\max_{\mathbf{G}} |E\{\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{g}(\mathbf{r})(\mathbf{G}^H\mathbf{g}(\mathbf{r}))^*\}|$$

$$= \arg\max_{\mathbf{G}} |\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{G}|\,, \tag{46}$$

$$\text{subject to } \mathbf{G_g}^H\mathbf{w}^{(0)}_{||\text{R-MF}||} = 0 \text{ and } \mathbf{G_g}^H\mathbf{G_g} = 1\,.$$

The magnitude cross-correlation criterion function $|\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{G_g}|$ and the orthonormality constraints are all phase invariant. So, to avoid unnecessary ambiguities and without loss of generality, we can identify the unique auxiliary vector that is a solution to our constraint optimization problem and makes $\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{G_g}$ real non-negative ($\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{G_g} \geq 0$). This is the vector

$$\mathbf{G_g} = \frac{\mathbf{R_g}\mathbf{w}^{(0)}_{||\text{R-MF}||} - (\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{w}^{(0)}_{||\text{R-MF}||})\mathbf{w}^{(0)}_{||\text{R-MF}||}}{||\mathbf{R_g}\mathbf{w}^{(0)}_{||\text{R-MF}||} - (\mathbf{w}^{(0)H}_{||\text{R-MF}||}\mathbf{R_g}\mathbf{w}^{(0)}_{||\text{R-MF}||})\mathbf{w}^{(0)}_{||\text{R-MF}||}||}. \tag{47}$$

We may comment at this point that the fact that no matrix inversion is explicitly required or implicitly assumed is expected to result in superior performance in realistic small-data-record adaptive operation.

The Max-Cross-Correlation (MCC) approach can be generalized for processing with multiple auxiliary vectors. In particular, we propose to consider conditional optimization, i.e. to determine the new vector, say $\mathbf{G}_2$, and the new scalar, say μ_2, by assuming that the main direction branch $\mathbf{w}^{(0)}_{||\text{R-MF}||}$ (Figure 3) has been updated and fixed to $\mathbf{w}^{(0)}_{||\text{R-MF}||} - \mu_1\mathbf{G}_1$, where $\mathbf{G}_1$ and μ_1 were determined by (47) and (45). These general algorithmic developments are summarized in the following Proposition.

Proposition 2 *Given the nonlinearity* $\mathbf{g}(\cdot)$ *then:*

(i) The maximum cross-correlation linear tap-weight filter $\mathbf{w}^{(0)}_{\text{PROP}}$ *is given by*

$$\mathbf{w}^{(0)}_{\text{PROP}} \triangleq \mathbf{w}^{(0)}_{||\text{R-MF}||} - \mu_{\mathbf{g}}\mathbf{G_g}, \tag{48}$$

where

$$\mu_{\mathbf{g}} = \frac{\mathbf{G}_{\mathbf{g}}^{H}\mathbf{R}_{\mathbf{g}}\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}{\mathbf{G}_{\mathbf{g}}^{H}\mathbf{R}_{\mathbf{g}}\mathbf{G}_{\mathbf{g}}}, \tag{49}$$

and

$$\mathbf{G}_{\mathbf{g}} = \frac{\mathbf{R}_{\mathbf{g}}\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - (\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\mathbf{R}_{\mathbf{g}}\mathbf{w}^{(0)}_{\|\text{R-MF}\|})\mathbf{w}^{(0)}_{\|\text{R-MF}\|}}{\|\mathbf{R}_{\mathbf{g}}\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - (\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\mathbf{R}_{\mathbf{g}}\mathbf{w}^{(0)}_{\|\text{R-MF}\|})\mathbf{w}^{(0)}_{\|\text{R-MF}\|}\|}. \tag{50}$$

(ii) The generalized multiple auxiliary vector linear post-processor is given by ($1 \leq P < L + M - 1$)

$$\mathbf{w}_{\text{PROP-Mul}} = \mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{P} \mu_i \mathbf{G}_i, \tag{51}$$

where $\{\mathbf{G}_i\}$ and $\{\mu_i\}$ are determined recursively as follows:

$$\mathbf{G}_{p+1} = \frac{\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i) -}{\|\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i) -}$$

$$\frac{-\,[\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i)]\mathbf{w}^{(0)}_{\|\text{R-MF}\|} -}{-\,[\mathbf{w}^{(0)H}_{\|\text{R-MF}\|}\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i)]\mathbf{w}^{(0)}_{\|\text{R-MF}\|} -}$$

$$\frac{-\,\sum_{j=1}^{p}[\mathbf{G}_j^{H}\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i)]\mathbf{G}_j}{-\,\sum_{j=1}^{p}[\mathbf{G}_j^{H}\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i)]\mathbf{G}_j\|}, \tag{52}$$

and

$$\mu_{p+1} = \frac{\mathbf{G}_{p+1}^{H}\mathbf{R}_{\mathbf{g}}(\mathbf{w}^{(0)}_{\|\text{R-MF}\|} - \sum_{i=1}^{p}\mu_i\mathbf{G}_i)}{\mathbf{G}_{p+1}^{H}\mathbf{R}_{\mathbf{g}}\mathbf{G}_{p+1}}, \tag{53}$$

for $1 \leq p < L + M - 2$, and $\mathbf{G}_1$, μ_1 given by (50) and (49), respectively. $\square$

VI. Simulation Studies

The optimization of the filter parameters follow the general principles of MSE-BER (Type I) optimization of Section IV.1 The corresponding algorithm is generalized below for the multipath case. In particular, let the receiver output be denoted by $u(\mathbf{r}, c)$, i.e. $u(\mathbf{r}, c) \triangleq sgn(Re[\mathbf{w}_c^H \mathbf{g}_c(\mathbf{r})])$ where c is the cutoff parameter of the non-linearity involved (eg. puncher or clipper type). Then, the distortion measure $\rho(\mathbf{r}_0, \mathbf{r}_1; c)$ defined at the receiver output by

$$\rho(\mathbf{r}_0, \mathbf{r}_1; c) \triangleq \frac{1}{2}\{\pi_0[1 + u(\mathbf{r}_0, c)] + \pi_1[1 - u(\mathbf{r}_1, c)]\}, \tag{54}$$

where $\pi_0 = \pi_1 = 1/2$ are the a priori probabilities of hypothesis H_0 ($b_0 = -1$) and H_1 ($b_0 = +1$), attains an average value equal to the BER, i.e.

$$P_e(c) = E\{\rho(\mathbf{r}_0, \mathbf{r}_1; c)\}, \tag{55}$$

where $\mathbf{r}_0$ and $\mathbf{r}_1$ correspond to received data vectors from H_0 and H_1 respectively. Exploiting the property in (55), we can develop a stochastic gradient technique that adjusts the cutoff parameter c to minimize the probability of error at the output of the receiver, as follows:

$$c_{n+1} = c_n - \alpha_n x_n(c_n), \tag{56}$$

where

$$x_n(c) \triangleq \frac{1}{2d_n}[\rho(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; c + d_n) - \rho(\mathbf{r}_{0,n}, \mathbf{r}_{1,n}; c - d_n)], \tag{57}$$

and $\{d_n\}$ and $\{\alpha_n\}$ are monotonically decreasing sequences of positive numbers such that $\sum \alpha_n = \infty$, $\sum \alpha_n d_n < \infty$, and $\sum \alpha_n^2 d_n^{-2} < \infty$. A similar gradient alogrithm can be obtained when more than one cutoff parameters are involved in the non-linearity used (eg. Hampel-type).

We examine DS-SS transmissions in the presence of 5 SS interfering signals and impulsive noise. The normalized signature cross-correlation of the interfering signals with the signal of interest is approximately 25% and the interfering signal energies are equal to 6, 7, 8, 9, 10dB. The system processing gain is 63. Figure 4 emphasizes the BER performance of the proposed receiver (implemented with one and two auxiliary vectors) and the MVDR receiver as a function of the norm of the projection of the SS signal of interest into the interfering subspace, when the clipper-type or puncher-type non-linear processing front-end is employed. Receiver needed ensemble averages are replaced by sample-average estimates based on a data-record of 150 samples. The channel noise is modeled according to the the outlier model, described in the following paragraph with the difference that no multipath is considered for this figure (ie. all functions are real).

The multipath fading channel is modeled as having 4 paths with zero mean complex Gaussian fading coefficients of variance 0.5 (i.e. $E\{|c_{k,m}|^2\} = 0.5$) for all paths and SS signals. The impulsive channel noise is modeled according to the ϵ-mixture disturbance model given by the following expression:

$$f_\epsilon(x) = (1-\epsilon) f_0(x) + \epsilon f_1(x) . \tag{58}$$

where $\epsilon \in [0,1]$ accounts for the probability under which the noise is $f_1(\cdot)$ distributed. The nominal pdf $f_0(\cdot)$ is taken to be complex Gaussian with variance $\sigma_0^2 = 1$. The "contaminating" pdf $f_1(\cdot)$ is assumed to be either a complex Gaussian pdf with variance $\sigma_1^2 = \gamma^2 \sigma_0^2$ ($\gamma^2 = 1.000$) or a delta function at $\pm\infty \pm j\infty$. The resulting model is the Gaussian ϵ-mixture model and the outlier model, respectively.

We compare the BER performance of the proposed receiver (implemented with one and two auxiliary vectors) with that of the MVDR receiver when the clipper-type, Hampel-type, or puncher-type non-linear processing front-end is employed. The BER of the conventional MVDR filter is also included as a reference point. The results are averages over 100 independent channel realizations.

In Figure 5 we present the BER performance of all receivers as a function of the energy of the SS signal of interest for the Gaussian ϵ-mixture model. Figure 6 carries out the same studies for the outlier model. For both figures, receiver needed ensemble averages are replaced by sample-average estimates based on a data-record of 150 samples. Figures 7 and 8 plot the BER as a function of the size of the data record for the same set-up as in Figures 5 and 6, respectively. That is, Figure 7 shows the BER performance for the Gaussian ϵ-mixture model, while Figure 8 deals with the outlier model.

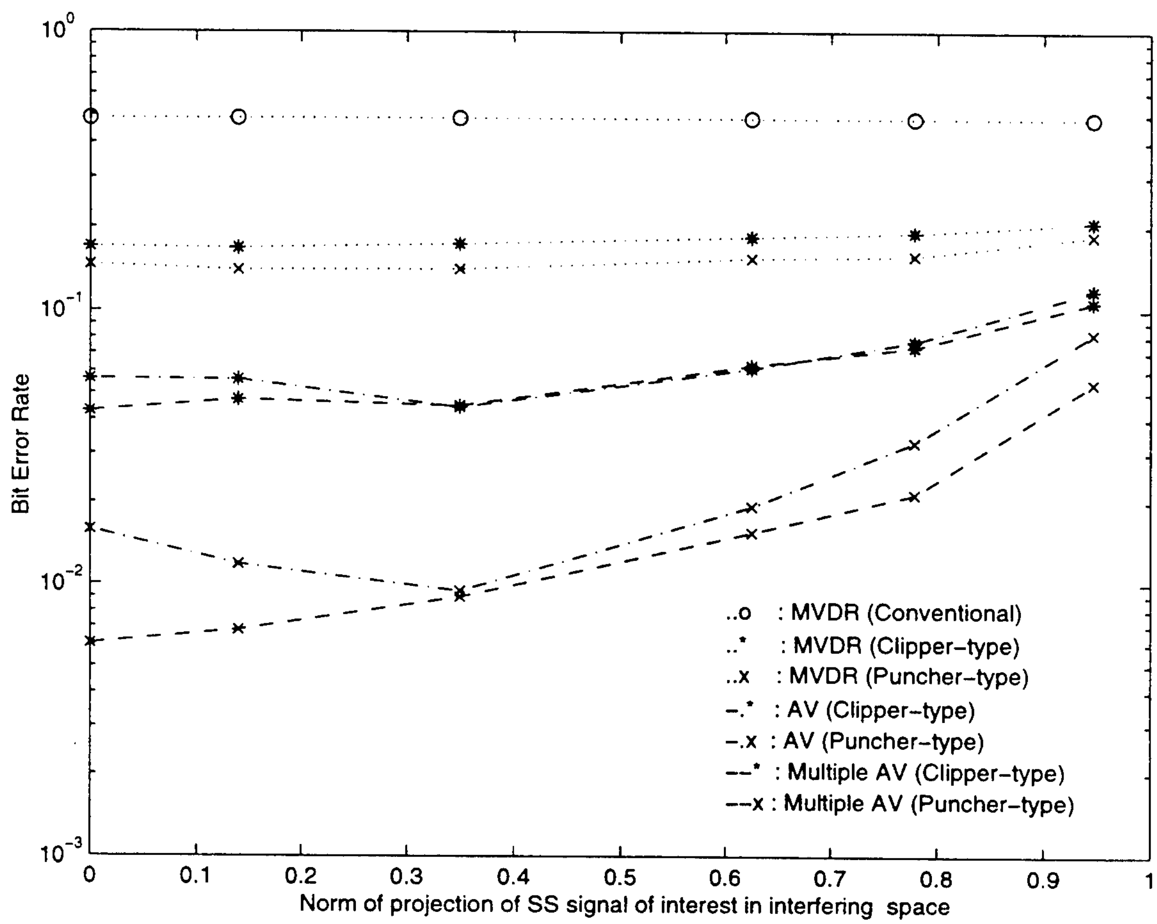


Figure 4: Bit error rate as a function of the norm of the projection of the SS signal of interest into the interfering subspace in the presence of outlier noise.

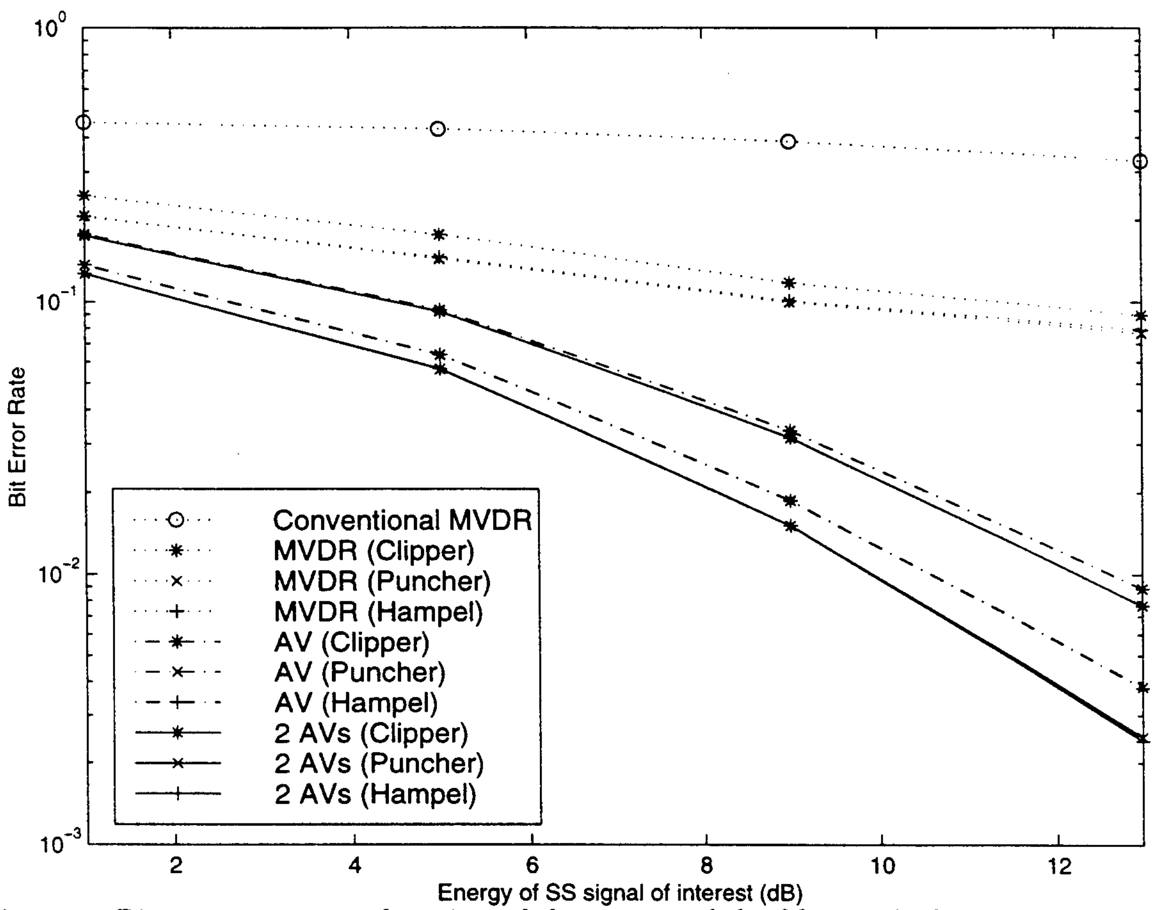


Figure 5: Bit error rate as a function of the energy of the SS signal of interest in the presence of ϵ-mixture noise.

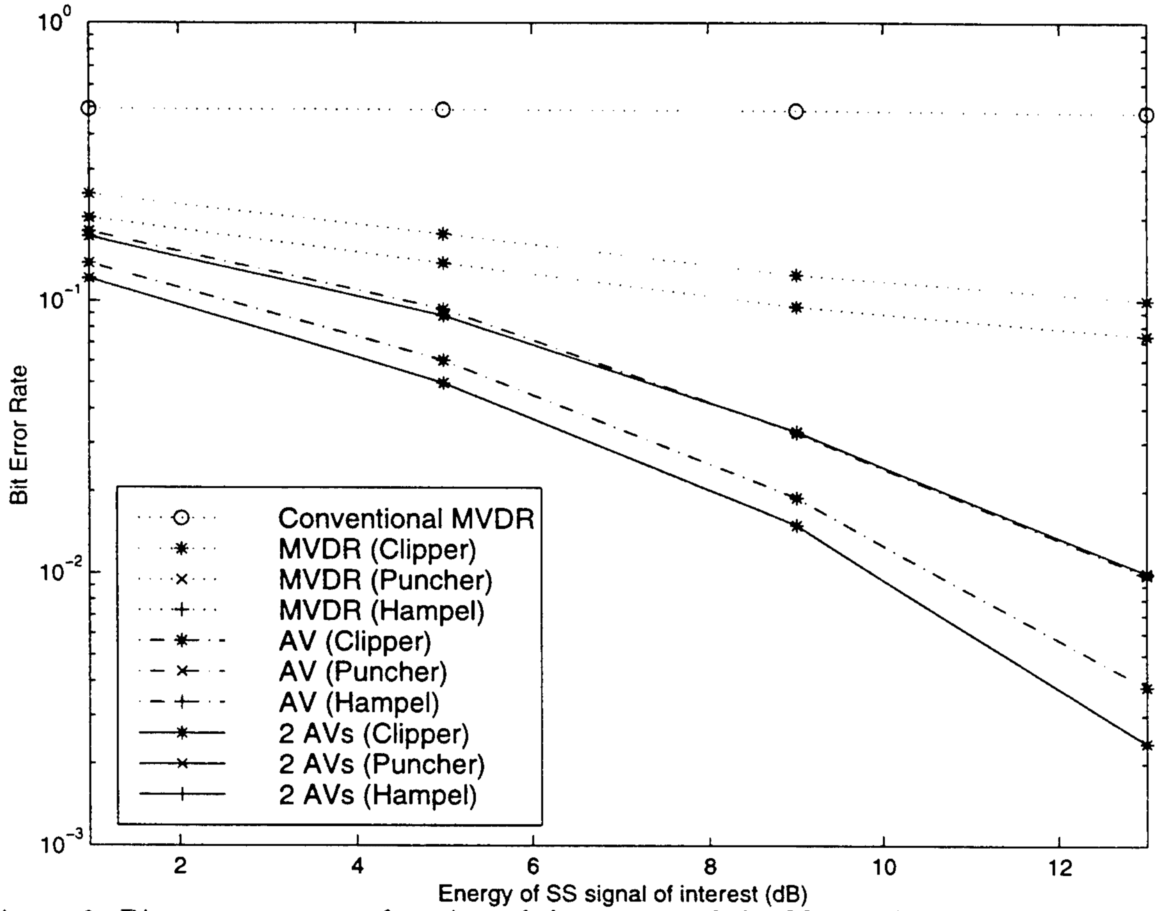


Figure 6: Bit error rate as a function of the energy of the SS signal of interest in the presence of outlier noise.

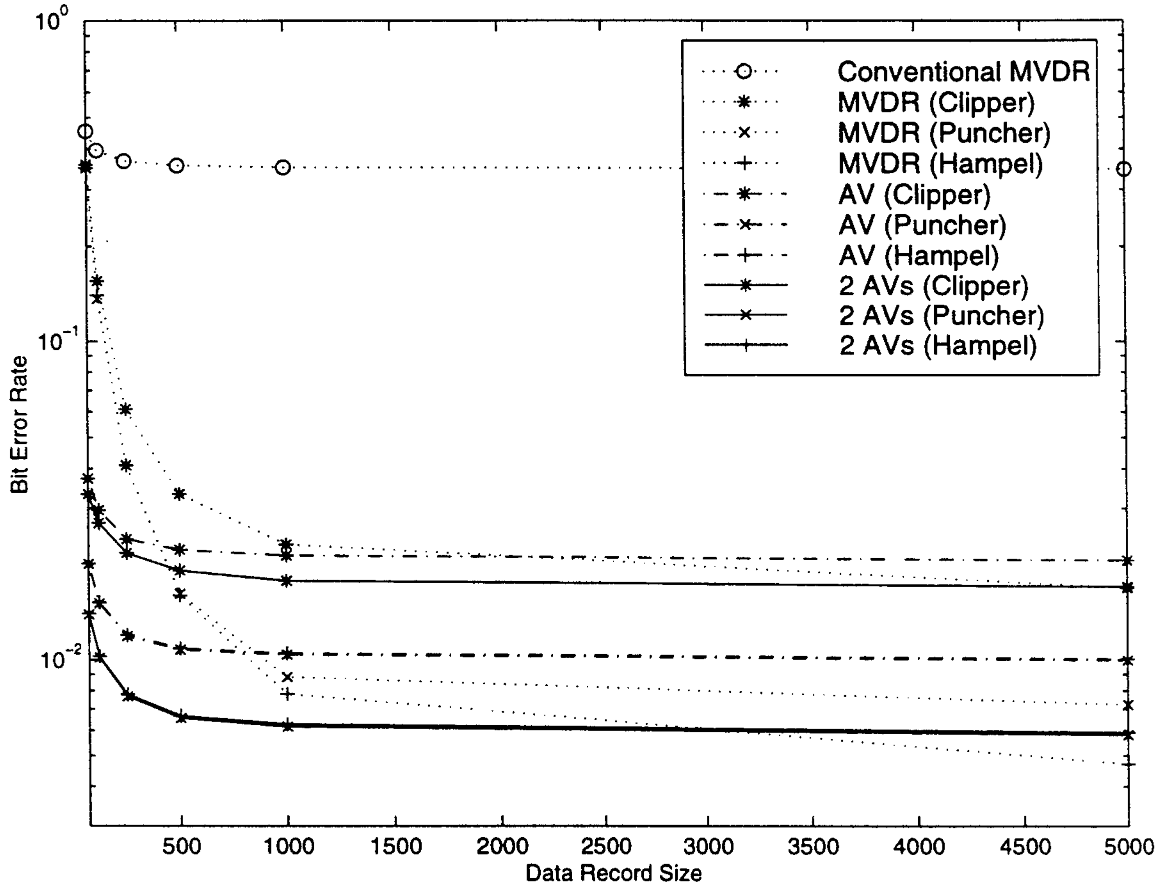


Figure 7: Bit error rate as a function of the training set size in the presence of ϵ-mixture noise.

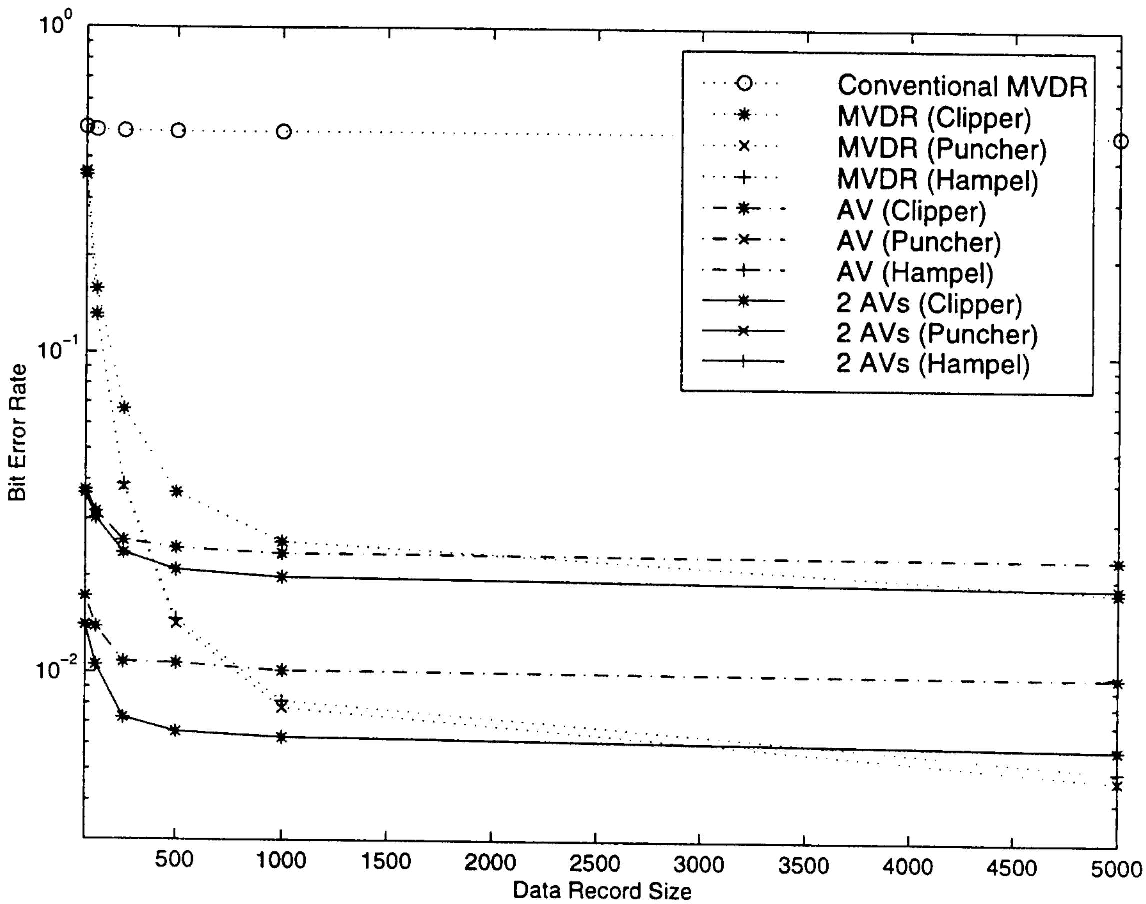


Figure 8: Bit error rate as a function of the training set size in the presence of outlier noise.

References

[1] A. D. Spaulding and D. Middleton, "Optimum reception in an impulsive interference environment-Part I: Coherent detection", *IEEE Trans. Comm.*, pp. 910-923, Sept. 1977.

[2] R. S. Blum, "Locally optimum distributed detection of correlated random signals based on ranks", *IEEE Trans. Info. Th.*, pp. 931-942, May 1996.

[3] A. D. Spaulding, "Locally optimum and suboptimum detector performance in a non-Gaussian interference environment", *IEEE Trans. Commun.*, vol. COM-33, No. 6, pp. 509-517, June 1985.

[4] G. A. Tsihrintzis and C. L. Nikias, "Performance of optimum and suboptimum receivers in the presence of impulsive noise modeled as an α-stable process", *IEEE Trans. Comm.*, pp. 904-914, Apr. 1995.

[5] A. B. Martinez, P. F. Swaszek, and J. B. Thomas, "Locally optimal detection in multivariate non-Gaussian noise", *IEEE Trans. Info. Th.*, pp. 815-822, Nov. 1985.

[6] B. Aazhang and H. V. Poor, "Performance of DS/SSMA communications in impulsive channels-Part II: Hard-limiting correlation receivers", *IEEE Trans. Comm.*, pp. 88-97, Jan. 1988.

[7] B. Aazhang and H. V. Poor, "An analysis of nonlinear direct-sequence correlators", *IEEE Trans. Comm.*, pp. 723-731, July 1989.

[8] S. N. Batalama, Final Report, 1997 Summer Faculty Research Program, US AFOSR, pp. 3-1 – 3-20.

[9] S. N. Batalama, M. Medley, and I. N. Psaromiligkos "Adaptive Robust Spread-Spectrum Receivers", *IEEE Trans. Comm.*, accepted for publication.

[10] A. Kansal, S. N. Batalama, and D. A. Pados, "Adaptive maximum SINR RAKE filtering for DS-CDMA multipath fading channels," *IEEE J. Select. Areas Commun.*, to appear Jan. 1999.

HIERARCHICAL- ADAPTIVE IMAGE SEGMENTATION

N. Bourbakis
Professor,
Dept. of Electrical Engineering
Image-Video Processing & Machine Vision Lab,

Binghamton University
Vestal Parkway,
Binghamton NY, 13902

Final Report for:
Summer Faculty Research Program
Armstrong Laboratory

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

And

Armstrong Laboratory

December 1998

HIERARCHICAL- ADAPTIVE IMAGE SEGMENTATION

N. Bourbakis
Professor,
Dept. of Electrical Engineering
Image-Video Processing & Machine Vision Lab,

Abstract

This paper presents a software methodology for near real-time and real-time user driven segmentation of color images or sequence of images (video) by using a low cost approach (a low cost camera and a PC). The segmentation approach used here is based on fuzzy-like reasoning and a color reflection. The methodology provides a human–like perception modeling in combination with fast smoothing, region growing and edge detection. Results on several images are provided to demonstrate the methodology's adaptive segmentation capabilities based on the user's requests.

Keywords: Segmentation of Color Images, Adaptive and Real-time Segmentation of Sequence of images

Acknowledgement : This work is partially supported by an AFRL research grant 1998

HIERARCHICAL- ADAPTIVE IMAGE SEGMENTATION

1. INTRODUCTION

Segmentation is one the most important processing or preprocessing step required by almost any image processing and vision task performed by a machine [1-18]. It is a process that has not a perfect solution for an input image if previous knowledge is not available. Thus, a number of special or general purpose approaches have been presented by scientists and practitioners to provide solve problems in different areas, such short or long distance robotic vision, ATR, visual inspection in production lines, pattern recognition, OCR, document processing, etc. [19-25]. Recently, research studies on image segmentation make use of previous knowledge [26,27], or attempted to extract knowledge by segmented a great variety of images under different resolution [28]. The effort of this paper belongs to the latter category, where the authors attempt to extract knowledge and classify it in various categories and applications. More specifically, they have developed a software tool-methodology, able to segment an image or a sequence of images in real-time, depended on the size of the images, under various conditions of sensitivity controlled by the user with a variety of parameters, such as smoothing, local contrast deviation, region filling, similarity deviation, region growing. The methodology runs on a PC under Windows 95 or NT.

2. FAST EDGE PRESERVING SMOOTHING

Smoothing is an important preprocessing step for any segmentation operation in that it allows for the reduction of noise within an image. Typically, areas of an image perceived to be a single solid color may actually be composed of pixels which span a broad range similar colors. Thus, a smoothing type operator, such as a low pass filter, may potentially make these areas more continuous in their color value while lessening the effects of salt and pepper noise. However, one problem with this type of filtering or averaging has always been the destruction or distortion (i.e. dilation) of edges within the image. A more desirable approach is to selectively smooth the image in a way that will preserve edges or areas of sharp contrast present within the image. Computation time is also a concern when developing such an approach since pixel-wise neighborhood operations performed on large images can be quite slow, precluding their use within real-time applications on general purpose (i.e. low-cost) hardware. These applications include real-time video processing for tasks such as target recognition or tracking.

The method presented here attempts to smooth an input image in a timely fashion while preserving the locations of edges within the image. This edge preserving smooth operation is accomplished by reducing the number or complexity of arithmetic operations performed on pixel neighborhoods and by employing the use of SIMD instructions that are now common on inexpensive general purpose microprocessors such as the Intel Pentium and Pentium II processors via MMX. The technique presented was adapted from earlier work described in [8,9] which produced robust methods for smoothing and segmenting images with disregard to computational complexity or time constraints. The method is applied using either 7 x 7 or 5 x 5 neighborhood windows and considers a pixel's local contrast before deciding if or how smoothing is performed (see figure 1). Color contrast between any two colors may be thought of as a value proportional to the Euclidean distance between those colors in the RGB color space (illustrated in figure 2). The overall algorithm is outlined in figure 1.

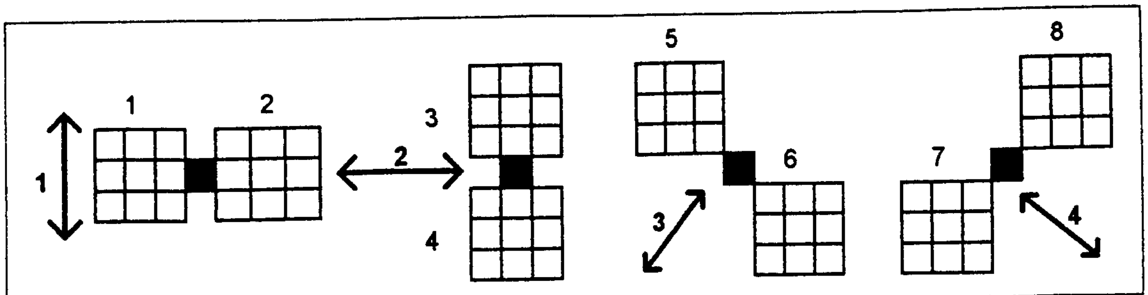
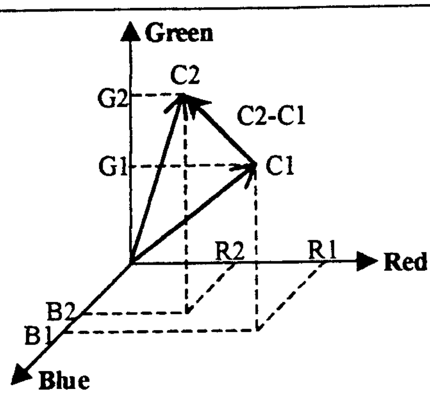


Figure 1. **Eight neighboring blocks of size 3×3 and four edge directions.** Unlike the original algorithm [], the standard *un-weighted* average of each 3x3 block is computed and used as the color value for contrast computations.



$$Contrast_{ij,b} = (R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2$$

Figure 2. **Two color vectors in the 3-D RGB color space and the computation of color contrast.** Local color contrast for pixel *(i,j)* is computed for each of the four edge direction (represented by the variable *b* in the equation above and illustrated in figure 1).

The entire smoothing algorithm is implemented entirely in assembly language making significant use of the MMX instruction set present on Intel Pentium family processors. To save time and avoid redundant computation, the 3x3 block averages (if a 7x7 window is chosen) for the entire image are computed and stored in a bitmap equal to the size of the original bitmap. Hence, when the average color value of a 3x3 block is needed, it is simply looked up within this bitmap rather than being computed on the fly. This also allows the input image to be overwritten with the output image without effecting block averages during processing.

In addition to least square deviations in RGB color space, color quantization is another technique employed here which effects the algorithm's propensity to smooth similar pixels that are adjacent to one another. Color values are quantized on values equivalent to powers of 2. This allows quantized values to be derived via a simple addition and shift operation. An addition of one half the quantization value is applied prior to the shift to guarantee that colors are quantized to the nearest quantized value. Additionally, since MMX registers and instructions are utilized, this operation can be performed in parallel for the red, green, and blue components of a color. It should also be noted that after quantization the color space is reduced (e.g. values ranging from 0-255 now range from 0-63 if a quantization value of 4 is chosen). This is important to consider when choosing a value for τ_{sm}, a threshold constant that is compared to the minimum and maximum color contrasts computed. This constant effectively discriminates pixels into one of three categories: noise pixels (min. and max. contrasts are both greater than τ_{sm}), edge pixels (min. contrast is less than τ_{sm} and max. contrast is greater than τ_{sm}), and interior pixels (min. and max. contrasts are both less than τ_{sm}). As shown in figure 3, the smoothing operation depends on the specific cases encountered.

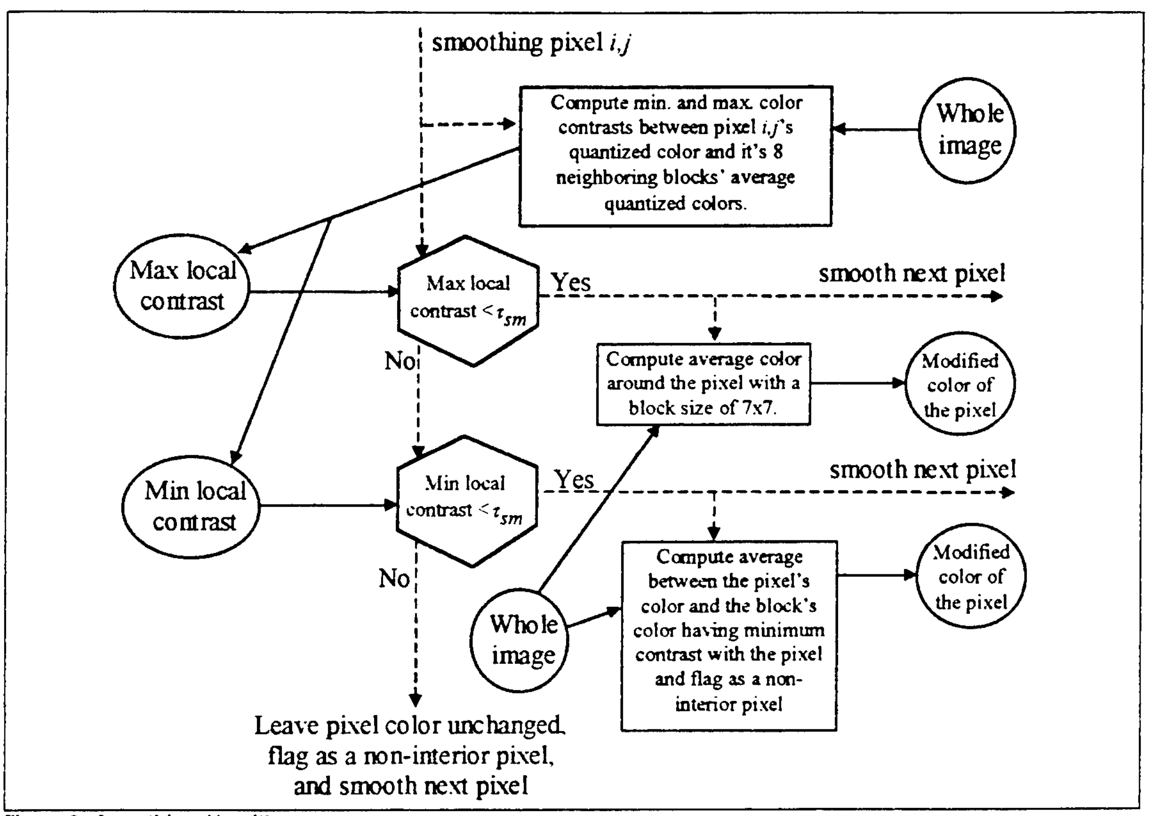


Figure 3. Smoothing Algorithm.

3. SEGMENT INITIALIZATION AND REGION GROWING

Initialize a set of segments for the image by performing efficient flood fill operations seeded at interior points (i.e. not flagged as non-interior, see figure 3). The fill algorithm used is described in [9] and is efficient in time and space requirements since it guarantees that each pixel within a formed segment is only evaluated a single time during the flood fill. The decision as to whether a given adjacent (4-connected) pixel should be filled during the flood operation is also based on its closeness in color to the original seed pixel for the segment. This similarity (or contrast) is again determined through quantization of color values and computation of the least square difference between the two colors in the RGB color space. As pixels are merged in segments they are encoded with an associated unique segment identifier. After the flood fill operation, most pixels within the image now belong to particular segments. Those pixels which have not been merged with any segment (i.e. are not encoded with a segment identifier) are merged through a process of region growing described below.

In order to merge the remaining pixels currently unbound to a particular segment, the edge pixels of the existing set of segments are propagated outward or grown. Then, as unbound pixels are encountered, they are merged with the closest segment of most similar color. This specific condition is calculated using a least squares difference of the red, green, and blue color components as well as a distance proportional to the distance from the original propagating edge pixel. This Euclidean distance is quickly approximated by computing the 3,4 distance transform (DT) during propagation (see figure 4). Propagation is terminated in a particular direction when color values are no longer being replaced in that direction. Clearly, this processing step also creates additional memory requirements by requiring memory for the storage of DT values within the bitmap and for a propagation stack used to keep track of color and DT values currently propagating.

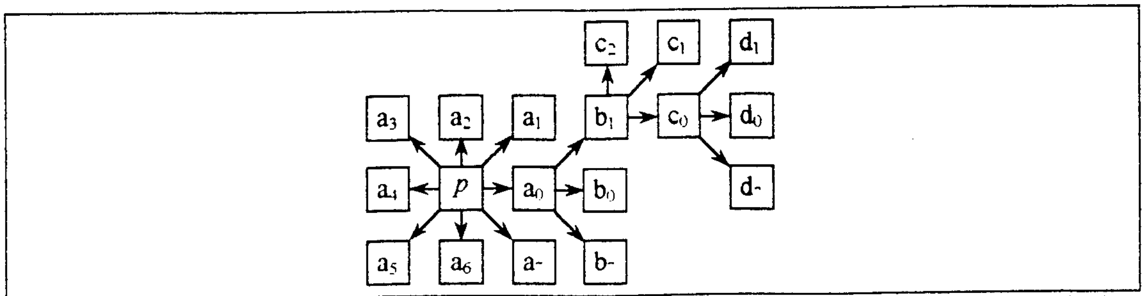


Figure 4. Edge Point Propagation. Shown is a simple example showing the propagation of a color value from skeletal point p where the only color value replaced is at point a_0. Thus, the color value and DT value at a_0 propagates to b_1, b_0, and b_7 where the only value replaced is at point b_1. Thus, the process continues again when the color value and DT value at c_0 is replaced.

4. TEST APPLICATION

The described segmentation procedure was implemented for Windows 95 and Windows NT and allows easy evaluation of results and experimentation with various parameters of the approach. Multiple images may be opened adjacent to each other and processed with varied parameter settings. Figure 5 shows one screen shot where the user may adjust these parameters.

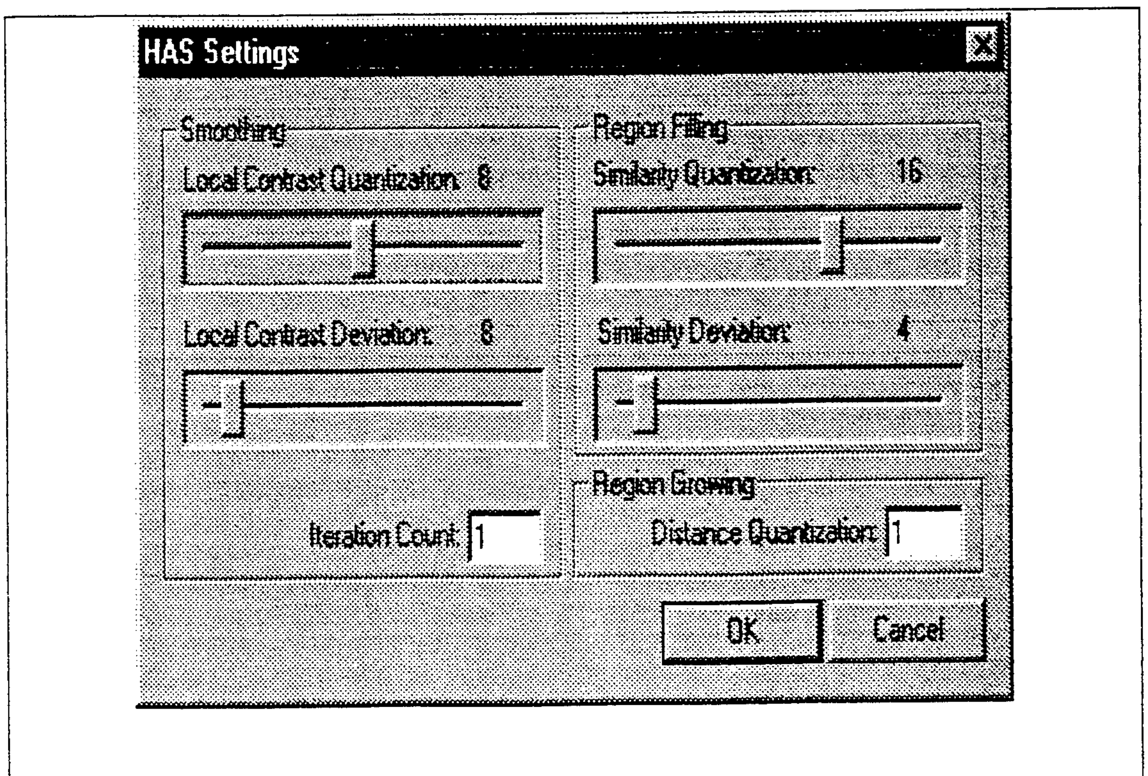


Figure 5. Modifiable Parameter Dialog Box.

Figure 6. Test Application Screen Shot. The top window displays the original image while the lower two windows display segmentation results when different parameter choices are made.

5. PERFORMANCE RESULTS

Performance of the methods presented here was evaluated by measuring the time requirements of each processing step. These timings were precisely obtained by using the profiling registers built into Intel Pentium family processors. Timing results for smoothing and region growing are listed in CPU cycle counts.

Image	Image size in pixels (WxH)	Smoothing (2 processors)	Flood fill and region growing	Segmentation method of [8]
Lena	512 x 512	28,629,858	86,044,047	>> 5 minutes
Peppers	512 x 512	28,934,578	75,440,006	>> 5 minutes
Video 1	320 x 240	7,201,471	24,337,244	>> 1 minute
Video 2	320 x 240	7,106,629	27,121,800	>> 1 minute

Table 1. Windows NT Performance results on a workstation with dual Pentium II 300Mhz processors.

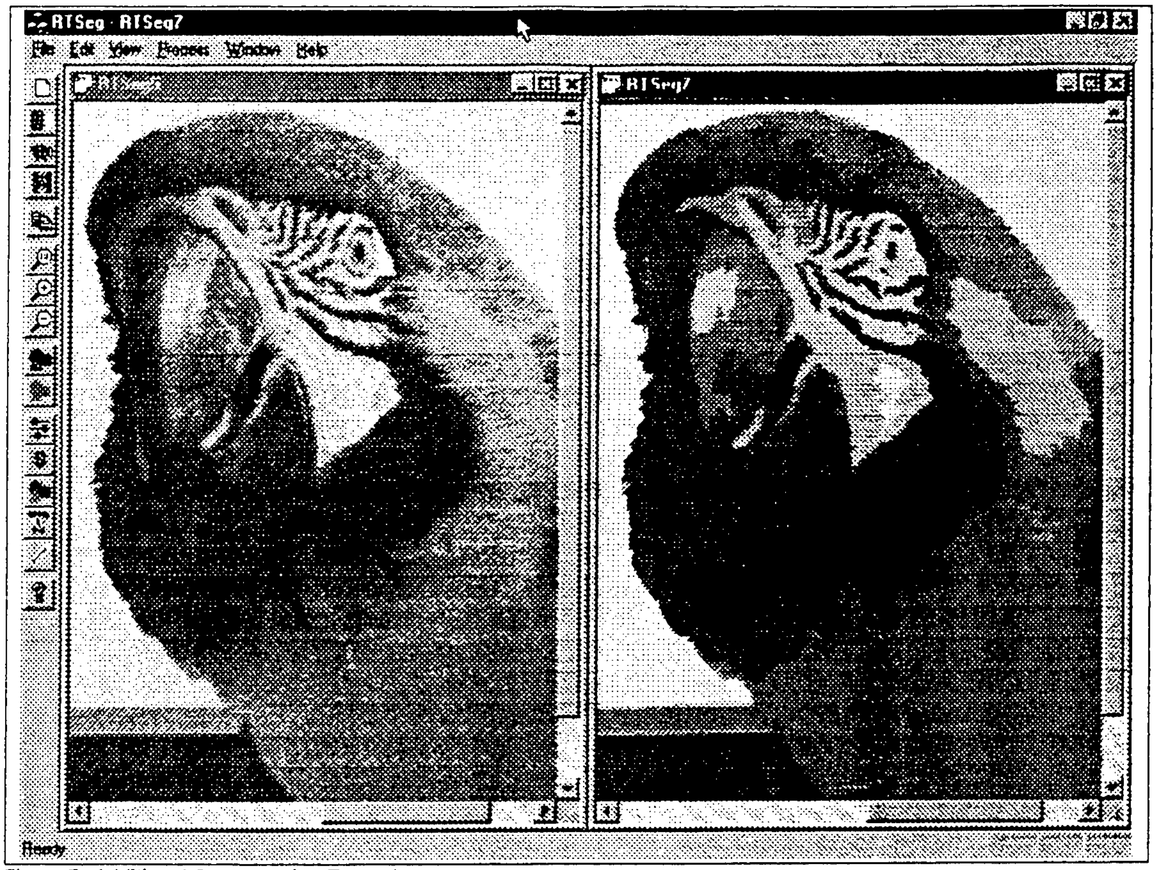


Figure 7. Additional Segmentation Example.

6. CONCLUSIONS

In this paper a tool-methodology, for adaptive segmentation of color images, or a sequence of images in real-time was presented. This effort was supported by an AFRL-Rome Lab research grant to develop a tool for extraction knowledge from segmented images with application to ATR in video images. The results are promising for real-time segmentation and near real-time segmentation for video with large size images.

References

[1] A.Ronsefeld, Survey on Techniques in Image Processing, CVGIP, 1993

[2] R. Schettini, "Low-Level Segmentation of Complex Color Images", *Signal Processing VI: Theories and Applications*, pp. 535-538, 1992.

[3] Nikhil R. Pal, and Sankar K. Pal, "A Review on Image Segmentation Techniques", *Pattern Recognition*, Vol. 26, No. 9, pp. 1277-1294, 1993.

[4] S. U. Lee, and S. Y. Chung, "A Comparative Performance Study of Several Global Thresholding Techniques for Segmentation", *Computer Vision, Graphics, and Image Processing*, Vol. 52, pp. 171-190, 1990.

[5] C. Huang, T. Cheng, and C. Chen, "Color Images' Segmentation Using Scale Space Filter and Markov Random Field", *Pattern Recognition*, Vol. 25, No. 10, pp. 1217-1229, 1992.

[6] F. Perez and C. Koch, "Toward Color Image Segmentation in Analog VLSI: Algorithm and Hardware", *International Journal of Computer Vision*, Vol. 12, No. 1, pp. 17-42, 1994.

[7] Y.Lim and S.Lee, "On the color image segmentation algorithm based on thresholding and fuzzy c-means techniques", Pattern Recognition J. 23, 3,935-952,1990

[8] A. Moghaddamzadeh, N. G. Bourbakis, "A fuzzy region growing approach for Segmentation of color images", PR Society *Journal of Pattern Recognition*, vol..30, 6,1997.

[9] A.Moghaddamzadeh, D.Goldman and N.Bourbakis, "Segmentation of color images with highlights and shadows using a fuzzy-like reasoning, Pattern recognition Letters, 1999

[10] M. Celenk, "A color clustering technique for image segmentation", *Computer Vision, Graphics, and Image Processing*, Vol. 52, pp. 145-170, 1990.

[11] R. Bajcsy, S. W. Lee, and A. Leonardis, "Color Image Segmentation and Color Constancy', *SPIE, Perceiving, Measuring and Using Color*, Vol. 1250, 1990.

[12] J.Liu and Y.H.Yang, "Multiresolution image segmentation", IEEE T-PAMI16, 689-700, 1994

[13] G. J. Klinker, "A Physical approach to color image understanding", *A. K. Peters*, Wellesley, Massachusetts, 1992.

[14] Andrew E. K. Pienkowski, "Artificial colour perception using fuzzy techniques in digital image processing", *TUV Rheinland*, Germany, 1989.

[15] Y. Xiaohan and J. Ylajaaski, "Image segmentation combining region growing and edge detection", *11th International Conference on Pattern Recognition*, Vol. III, The Netherlands, 1992.

[16] M. Chapron, "A New Chromatic Edge Detector Used for Color Image Segmentation", Proceedings, 11th IAPR International Conference on Pattern Recognition, Vol. 3, Conference C: Image, Speech, and Signal Analysis (ICPR11, The Hague, The Netherlands, August 30 – September 3, 1992), IEEE Computer Society Press, Los Alamitos, California, pp. 311-314, 1992.

[17] M. Pietikainen and D. Harwood, "Edge Information in Color Images Based on Histograms of Differences", Proceedings, 8th International Conference on Pattern Recognition (Paris, France, October 27-31, 1986), IEEE Publ. 86CH2342-4, pp. 594-596, 1986.

[18] A.D.Ventura, C.Motta and R.Schettni,"A color-driven image segmentation algorithm", Proc. 6th Int. Conf. On Image Analysis Processing, 1992

[19] Songhua et. al . "Target discrimination and recognition using high resolution features", NAE Conf. 280-286, 1992

[20] T. Pavlidis and J.Zhou,"Page segmentation and classification", CVGIP 54,6,1992

[21] K.Etemad and R. Chellappa, "Separability based multiscale basis selection and feature extraction from signal and image classification", IEEE T-IP, 7,10, 1453-65, 1998

[22] F.Moscheni, S.Bhattacharjee and M.Kunt, "Spatiotemporal segmentation based on region merging", IEEE T_PAMI, 20,9,1998

[23] S.Dickkinson, D.Metaxas, A.Pentland, "The role of model based segmentation in the recovery of volumetric parts from range data", IEEE T-PAMI, 19, 3,1997

[24] C.Debrunner and J.Ahuja, "Segmentation and factorization based motion and structure estimation for long image sequences", IEEE T-PAMI, 20, 2, 1998

[25] R.Stiefelhagen, J.Yang and A.Waibel,"A model based gaze tracking system", Int. Journal on AIT, vol.6, 2, 1997

[26] M.Jager, C.Tresp, M.Moser and J.Hiltner,"Image analysis based on fuzzy knowledge", Int. Journal on AIT, vol.7,1,1998

[27] T.Hofmann, J.Puzicha, J.Buhmann, "Unsupervised texture segmentation in deterministic annealing framework", IEEE T-PAMI, 20,8, 1998

[28] N. Bourbakis, "Extracting knowledge and Statistics from Segmented Images", TR-1997 , pages 17, Int. Journal AIT. 1999

INFORMATION FUSION WITH MULTIPLE FEATURE EXTRACTORS FOR AUTOMATIC TEXT CLASSIFICATION AND RETRIEVAL – SCALING ISSUES

Venu Dasigi[1]
Associate Professor
Department of Computer Science
Southern Polytechnic State University
1100 S. Marietta Parkway
Marietta, GA 30060-2896
vdasigi@spsu.edu

Final Report for:
Summer Research Extension Program

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

January, 1999

[1] Also a member of the Rome Laboratory Information Institute.

INFORMATION FUSION WITH MULTIPLE FEATURE EXTRACTORS FOR AUTOMATIC TEXT CLASSIFICATION AND RETRIEVAL – SCALING ISSUES

Venu Dasigi
Associate Professor
Department of Computer Science
Southern Polytechnic State University

Abstract

We report here on our recent experiments, and results on automatic classification of free text documents into a given, *large* number of categories. Text classification has applications in information filtering and routing, automatic indexing, retrieval, etc. We use a latent semantic indexing-based feature extractor and integrate neural net learning into the method. We then compare the performance with our past experiments of classification into a smaller number of categories. We also conducted experiments to assess the impact of the training set size in this context.

The feature extractors are based on the "latent semantics" of a *reference library*. Intuitively, the technique of latent semantic indexing (LSI) [Deerwester, et al., 90] projects any document into a dimensionally reduced space of concepts from the reference library. Neural nets are used to incorporate a learning component, as well as to fuse information from different combinations. Metrics such as micro and macro averaged precision, recall and correctness are used to compare the performance of the different feature extractors and effectiveness of their fusion.

The preliminary results indicated that the problem of classification into a large number of categories is substantially more complex. So, we focused exclusively on this issue, and evaluated the impact of training set size on performance. This is an important and new direction of experimentation, which was indicated by earlier experiments. Allowing for the inherent complexity of the problem, neural net training seemed to have added only a little value to straight LSI-based results. A surprising observation was that even substantial variation in the training set size had little impact on the results. We discuss possible explanations for this surprising behavior.

INFORMATION FUSION WITH MULTIPLE FEATURE EXTRACTORS FOR AUTOMATIC TEXT CLASSIFICATION AND RETRIEVAL – SCALING ISSUES

Venu Dasigi

1. Introduction

Information *retrieval* and *classification* refer to automated retrieval in response to a query and automated classification into various categories of what may be thought of as *unstructured* information. This endeavor, therefore, is to be contrasted specifically with database retrieval, where the data are *structured.* In this age of information explosion, information retrieval and classification have become more critical research areas than ever. In this work, our focus is on automated classification of unstructured text. Such automated classification has many applications, such as information filtering based on user profiles, personalized information delivery, etc. For the problem of information retrieval itself, classification of information can help in automatic indexing, as well as in other ways. Other related applications include classification and routing electronic message traffic (e.g., in military aircraft operations), e-mail filtering (very useful when a department receives thousands of e-mail messages a day), etc.

Thus, our initial ***goal*** in this work was to identify some new "feature extractors" that extract important features from text, and also to evaluate the effectiveness of the individual feature extractors, and that of information fusion from multiple such feature extractors, in automatically classifying free text documents into a given number of categories. This was consistent with the prior work that we had done at the Air Force Rome Laboratory. During that work, we had also identified a couple of other important issues. First, *scaling the approach to a larger number of categories* has been an important concern. Since the Reuters data lends itself to experiments on classification into a larger number of categories, we focused on this issue. Neural net learning was incorporated into the approach so that information derived by a feature extractor from training documents, along with the correct classification, are used to train the neural network. With neural net training in place, we returned to an issue that was indicated by our previous experimentation, namely, *evaluating the impact of training set size on performance*.

We start with the standard feature, namely, term frequency. We then project a document vector into the space of semantic concepts from a *reference library* using Latent Semantic Indexing (LSI) [Deerwester, et al., 90]. A term-document (or a phrase-document) *reference matrix* represents the frequencies of terms (or phrases, respectively) from a collection of documents from the reference library. In LSI, this large and sparse term-document matrix is reduced into three smaller matrices, one of which is simply a diagonal matrix, by singular value decomposition (SVD). The diagonal matrix yields the singular values from which the dominant ones are selected. Thus, instead of representing documents by thousands of possible terms or phrases, LSI allows for a document to be represented by a substantially smaller number of *factors* that intuitively capture the significant "latent semantics."

Thus, a reference matrix is the term-document matrix of a reference library / collection of documents. A reference library is the collection of documents that adequately represents all concepts of interest. The SVD computations are performed on such a reference library only once, and the linear transformations mentioned above essentially project *any* new document into the concept space represented by the reference library. One of the parameters we experimented with in our past work is the size of the reference library.

Section 2 describes background material and the general approach. It also discusses the metrics we use. Section 3 deals with the experimental set up. Section 4 summarizes and discusses the results. Section 5 concludes the paper by putting this work in the context of other related work. Appendices A through D include various tables that show details of performance for each experiment and other details.

2. Background

Our hypothesis in this work has been that a neural network, as in GRAIL [Uberbacher and Mural, 91] [Xu, et al., 94], would incorporate a learning component into a classification method, besides possibly integrating information from different features in a systematic way, thereby improving performance over previous approaches. To this end, our specific goal in this work has been to investigate the problem of classifying text documents into a large number of categories, measure the performance, and compare /contrast it against that involving a smaller number of categories. In order to make training the neural net practical, the primary issue is containment of the dimensionality of the pattern recognition problem, and Latent Semantic Indexing (LSI) [Deerwester, et al., 90] is used for this purpose. (Thus, there are two ways to view the approach: either as a neural net method, made practical by LSI, or as an LSI-based approach augmented with neural net learning.) We first present a quick overview of how LSI is used in this work and then briefly describe the metrics used to measure performance, adopted mostly from [Dasigi, 97].

2.1. Dimensionality Reduction with LSI

The developers of LSI indicate that a query may be viewed as a pseudo-document and may be represented as a vector of a chosen number of factors, so it may be placed in the same vector space as regular document vectors [Deerwester, et al., 90]. This technique is used in here to reduce the dimensionality of a regular document, so it can be input to a neural network. This is done as follows. First, as pointed out in the introduction, a reference term-document sparse matrix X is derived from the library of documents that are of interest. This matrix is split into three matrices by Singular Value Decomposition (SVD) [Forsythe, et al., 77], so that

$$X = T.S.D'$$

Here, X is a txd matrix, where t is the numbers of distinct terms (word roots) and d is the number of documents in the reference collection. The order of T is txk, that of S, which is a *diagonal* matrix, is kxk, and that of D is dxk, where k is the chosen number of factors. (LSI research indicates that a choice of k near 100 is good.) Now, the pseudo-document vector D_Q corresponding to a 1xt query vector Q may be derived simply as:

$$D_Q = Q.T.S^{-1}$$

In this work, we use this same idea to squash any 1xt document vector into a 1xk vector that serves as input to the neural network. The dimensionality reduction is substantial: t is generally a few thousand to a few tens of thousands, and k is near one hundred. *The only care that must be exercised is to make sure that the reference term-document matrix that is used as the starting point is one that "adequately" represents all concepts of interest.* (We return to address the adequacy issue in Section 4 on Experiments and Results.) This requirement is no more stringent than would be required in the standard LSI approach.

2.2. Performance Metrics

Precision and *recall* are very commonly used as measures of performance in information *retrieval*. Precision is defined as the fraction of retrieved items that are actually relevant (the rest are *false positives*), and recall is defined as the fraction of relevant items that are actually retrieved (the rest are *false negatives*). Thus, precision is inversely related to false positives and recall is inversely related to false negatives during retrieval.

Lewis is perhaps the first to define these metrics in the context of information *classification* [Lewis, 92]. In classification, there are multiple classes, zero or more of which are to be assigned to each document. If each class were viewed as a retrieval query, the definitions fall into place. That is, for any class, precision is the fraction of items assigned to the class that actually belong to the class, and recall is the fraction of items that belong to the class that are actually assigned to the class. We further define *correctness* of classification as the fraction of all documents that are classified correctly with respect to the class, namely, the ratio of the sum of the numbers of *true* *positives* and *true negatives* to the total number of documents. That is, for the i-th class, if tp_i, fp_i, tn_i and fn_i stand respectively for the numbers of true positives, false positives, true negative and false negatives, we have:

$$\text{precision}_i = tp_i / (tp_i+fp_i)$$
$$\text{recall}_i = tp_i / (tp_i+fn_i)$$
$$\text{correctness}_i = (tp_i+tn_i) / (tp_i+tn_i+fp_i+fn_i)$$

The picture gets a little more complex. With these metrics distributed over all the classes, to get global performance measures, some kind of averaging is needed. There are two ways to do the averaging. In *macro-averaging*, the individual measures for each class are taken and an overall average is computed simply by adding them and dividing the grand total by the number of classes. In *micro-averaging*, the grand total aggregates of the individual numbers for true/false positives and true/false negatives are computed and the appropriate ratios give the metrics. Thus, assuming below that i ranges over all classes in the summation, with the scope of Σ being the single term or parenthesized expression following it, and N is the number of classes, we have:

$$\text{macro-precision} = \Sigma_i \text{ precision}_i / N$$
$$\text{macro-recall} = \Sigma_i \text{ recall}_i / N$$
$$\text{macro-correctness} = \Sigma_i \text{ correctness}_i / N$$

$$\text{micro-precision} = \Sigma_i\, tp_i / \Sigma_i\, (tp_i+fp_i)$$
$$\text{micro-recall} = \Sigma_i\, tp_i / \Sigma_i\, (tp_i+fn_i)$$
$$\text{micro-correctness} = \Sigma_i\, (tp_i+tn_i) / \Sigma_i\, (tp_i+tn_i+fp_i+fn_i)$$

A simple analysis reveals that macro-averaging and micro-averaging always result in the same value for correctness. Another point to note is that the correctness measure can increase just by increasing true negatives, without directly affecting the other measures. Since generally the number of true negatives is very high, a high correctness measure can give a false / misleading sense of good performance.

Generally, in any retrieval / classification experiment, a single parameter is varied to get a precision-recall pair at each parameter value. Together, they form a precision-recall curve. Generally, in such experiments, when the parameter is changed so as to increase precision, the recall figure decreases and vice versa. In the extreme case, when the parameter is such that there are no false positives (100% precision), very likely we recall very little by being conservative. On the other hand, if the parameter is such that there are no false negatives (100% recall), we are so permissive that we retrieve everything, decreasing precision substantially.

Due to this inverse relationship between precision and recall, it is hard to compare the performance of different approaches by looking at isolated figures. The entire precision-recall curves need to be compared. Alternatively, a single measure called the *break-even point* is sometimes used for comparison. The break-even point is the point on the precision recall curve that has the same value for precision and recall. A higher break-even precision (recall) indicates better performance. The break-even point may be linearly interpolated from the two points closest to it on either side on the precision-recall curve. If the two points on either side of the break even point are $<p_1, r_1>$ and $<p_2, r_2>$ such that $p_1>r_1$ and $p_2<r_2$, the break-even point $<b, b>$ is given by:

$$b = (r_2*p_1 - r_1*p_2) / (r_2-r_1 + p_1-p_2)$$

3. Experimental Set Up

In order to train the neural network, we turn to a collection of data made available by the Reuters news agency, and prepared by Dr. David Lewis[2]. The Reuters data are described in [Dasigi et al., 97] and [Lewis, 92]. The advantage of this collection is that the Reuters stories are already manually classified, and include a sufficiently high number of stories per category. The Reuters stories have anywhere from zero to five categories assigned to each of them. Since there are about 135 categories for the Reuters data, we originally combined several of them into single "higher level" categories, as suggested by the data developers; our past results using these higher level categories were impressive, as reported in [Dasigi, 97]. The categories used now appear in an Appendix D; the higher level ones that we had used appear below:

1. *money / foreign exchange*
2. *shipping*
3. *interest rates*
4. *economic indicators*
5. *currency*
6. *corporate*
7. *commodities*
8. *energy*

---

[2] The Reuters-22173 text categorization collection is copyrighted by Reuters, Ltd., and is distributed freely *for research purposes only*. Copyright for additional annotations and a number of auxiliary files resides with David D. Lewis and the Information Retrieval Laboratory at University of Massachusetts. The authors thank the providers of the Reuters-22173 collection for making it available. It can be obtained at the URL *ftp://ciir-ftp.cs.umass.edu/pub/reuters1*.

The size of the Reuters corpus is about 25 MB, representing 22,173 Reuters news wire stories. The stories are pre-classified and pre-analyzed. Thus, in addition to the 25 MB of raw text, many additional files are provided that contain information such as what categories, if any, have been assigned to each document, what words are present in each document (story) with what frequency, what multi-word noun-phrases occur in each document with what frequency, etc. Many details about the collection can be found in [Lewis, 92], and also in the *README* file associated with the collection. Details of distribution of the documents among the various categories may be found in [Dasigi, et al., 97].

The collection contains 22,791 different words and 29,496 different noun-phrases indexed to the documents. These numbers are rather high, and part of the reason is "words" corresponding to the different numeric values that occur in the documents, special symbols, etc. have all been treated as genuine words. It is up to the system that processes this information to use them appropriately. (In our work, we do not do anything special in this regard.) The documents come with pre-assigned classification, with about 135 topic categories and another 539 other kinds of categories.

As mentioned before, the main feature extractor used in this work is based on LSI applied to word frequencies. The choice of the reference library can be important in LSI, because documents are represented in the space defined exclusively by the concepts from the reference library. The reference library that we used comprises the first 3,000 documents.

4. Experiments and Results

We start off by pointing out the precision and recall values when straight LSI-based retrieval is adopted to perform classification into the eight higher level categories – an easier problem. This is done by first using LSI to retrieve the document from the reference library that best matches the document to be classified. In other words, the document to be classified is used as if it were a query, as explained in [Deerwester, et al., 90]. Then the categories assigned to the best-matching document from the reference library are assigned to the "query document", completing the classification task. No learning is incorporated into this method. In this experiment, there are no parameters to vary as in the other experiments, and so there is a single set of macro- and micro- averaged precision and recall figures, and correctness. The results are as follows:

Macro-Precision	Macro-Recall	Micro-Precision	Micro-Recall	Correctness
9.55%	5.60%	20.86%	13.56%	89.76%

It may be noted that although the correctness figure is good, the precision-recall figures are not very high. One reason for this behavior might be that the number of true negatives is perhaps very dominating, pushing the correctness percentage very high. Since true negatives have no bearing on precision and recall, those figures are small. In particular, it may be noted that in either kind of averaging, both the precision and recall figures are rather low, a behavior that may be contrasted with the results from the remaining experiments that (i) incorporate learning and (ii) attempt classification into the full contingent of 135 categories, rather than into just eight.

The number of output units in all neural net experiments is 135, one for each class. Coincidentally, the maximum number of factors that could be used from the singular value decomposition (SVD) computations was also 135[3]. The neural nets therefore also have 135 input units. The number of hidden units used is 9 in most of the experiments. The specific architecture, namely the number of hidden layers (always one in our experiments) and the number of hidden units, is chosen after some experimentation, based on performance after initial training. (Several meaningful choices have been experimented with for this purpose. Specifically, we also report on some experimentation with a slightly larger number of hidden units, namely, 15, as discussed later.) All the networks use the hyperbolic tangent for transfer function and the delta rule for learning.

Of the 22,173 documents, the first 4,000 documents are initially used for training the neural network, and the remaining 18,173 are used for testing. The neural network is trained for 16,000 iterations at a time (each training document presented four times), all the way up to about 192,000 iterations. After each step of 16,000 iterations, the network is tested with all the training documents and then with all the test documents, separately, and the outputs saved in files. These raw output files are interpreted with the threshold value (that distinguishes the binary decision) as the parameter and macro- and micro-averaged precision and recall, as well as correctness, are calculated at each parameter value. Break-even points are also calculated for the macro-averaged precision-recall curve as well as the micro-averaged precision-recall curve. The performance measures presented here are those for the test data set that correspond to the same point of training that results in the best break-even points in the training data.

Then we repeat the experiments by increasing the training set size by the next 4,000 documents. In other words, the next 4,000 documents are moved from the test set to the training set. Thus, when 8,000 documents are used for training, only 14, 173 documents are used for testing. The other experiments use training set sizes of 12,000, 16,000, and 20,000, with test set sizes of 10,173, 6,173, and 2,173, respectively. These experiments are called experiments 1 through 5, with experiment 1 using 4,000 training documents and experiment 5 using 20,000 training documents.

The relevant results are distributed over several tables, which are included with the report. Summary tables with breakeven points have been created for each experiment and reported in Appendix A. Whenever additional information is required (e.g., correctness, precision-recall measures for other threshold points), it is included as appropriate. A preliminary analysis of the raw Reuters-22173 data shows that of the 22,173 documents 11664 (53%) documents have topics assigned to at least one category and 10509 documents were not assigned at all. The point to note here is that a significant portion of the data (47%) does not belong to any of the 135 categories. These results can be found in Appendix C.

As pointed out in the section on Performance Metrics, the raw data tables for macro/micro precision-recall results (Appendix B) show that macro-averaging and micro-averaging result in the same correctness values for all threshold values. For negative threshold values, correctness has small values (about 1%), recall has large values (about 100%) and precision has small (about 1%). For those thresholds values there are no false negatives, which implies that everything was retrieved, decreasing precision substantially. However, as the threshold value increases by .01, correctness improved, precision increases (up to 100%) while recall decreases and a breakeven point is noted at sometime. From macro/micro precision-recall results one could observe that there is clearly an inverse relationship between precision and recall.

For clarity, simplicity and comparison, the breakeven points for all the experiments were compiled into ten tables (see Appendix A). Each table contains an experiment number, the number of documents (training/testing), the number of learning iterations and the micro/macro measures (breakeven, threshold and correctness points). The results show that the macro-average breakeven points are in the range 6.14 - 16.44 for training results and 4.71 – 8.59 for testing. The micro-average breakeven points range from 28.12 to 36.82 for training and 31.58 to 33.53 for testing. A point to note is that a higher breakeven point indicates an improvement in both precision and recall. The best breakeven points ***for testing with training data*** came from experiment 1.

Experiment 1 Training – 4000 Testing – 18173 Performance when tested with training data

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
128000	**16.44**	0.02 – 0.03	96.29 – 97.29			
176000				**36.82**	0.13 – 0.14	99.52 – 99.56

However the results also show that when in Experiment 1, the network was tested with 18173 test documents the results were not the overall best of the five experiments. Better testing results came from experiment 2 –5 with experiment 5 showing the best macro-average and experiment 2 showing the best micro average. This observation

[3] The exact choice of this number is arbitrary; a number of about 100 is recommended by the developers of LSI.

is consistent with the expectation that a larger training set can positively influence the performance of the neural network for by the best macro-averaged results (Experiment 5). Since neural networks tend to be biased toward the more "populous" categories, it is perhaps more important to focus on the macro-averaged performance. However, the macro-breakeven points are too small to be sufficiently impressive. Interestingly, the best micro-averaged values occur in Experiment 2.

Experiment 5 Training – 20000 Testing – 2173 Performance with test data

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
144000	**8.59**	0.01 – 0.02	93.44 – 96.39

Experiment 2 Training – 8000 Testing - 14173 Performance with test data

Learning Iterations	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
144000	**33.53**	0.13 – 0.14	99.26 – 99.31

5. Conclusion

In micro-averaging calculations, all categories used in the classification are viewed equally important while in macro-average calculations the categories with more documents assigned to them dominate. Since neural networks can be over-trained or biased toward the more populous categories, the macro-average performance is a better measure of precision-recall in most categories, not just the specific or populous ones. The results of experiment 5 could be significant from this perspective; however, the macro-breakeven point values are somewhat small.

The overall percentages of correct result for all breakeven points in the experiments were very good, with correctness values ranges from 91.60 – 99.70 %. In preliminary analysis, this seems to demonstrate that the neural network performs the task of classification correctly for the known threshold point. However, these impressive numbers are attributable to a large number of true negatives (that is documents not belonging to a category are classified as such).

Extending the success of the approach from the original task of classification into 8 categories to the more complex task of classification into 135 categories proved to be very hard. The new problem is admittedly much harder, and the neural network requires very carefully controlled training. Neural network literature indicates that the size of the training set needs to be much larger, too. After several weeks of careful planning and experimentation, the performance was still not entirely satisfactory. The micro breakeven point is somewhat better than the performance obtained with straight LSI with just 8 categories, although such an achievement with the macro breakeven point would have been of better interest. A tentative conclusion that may be drawn at this time is that either more experimentation is needed to integrate neural network training with LSI or the neural network approach reaches its limitation as the complexity of the classification decision is increased.

At this point it was decided to change the architecture of the neural network so as to increase the number of hidden units from 9 to 15. This increase is more in concert with the recommendation from NeuralWorks User Manual. However, there was no substantial difference in performance. These tables for Experiments 1 and 5 for the revised architecture are also included in Appendix A, at the end, with appropriate headers.

To compare our approach itself with other similar methods, we believe our approach makes for a synergy between LSI and neural network learning. The dimensionality reduction by LSI limits the size of the neural network making it practical, and the neural network augments LSI with learning. Using a reference library with a one time SVD computation makes the approach practical. [Schütze, et al., 95] also employ an LSI-based approach in conjunction with neural network learning. Our approach differs from theirs in using a reference library and in employing multiple sensors. In their work, they seem to have used the set of all tested documents together as the reference library, obviating the need for a hidden layer in the neural network. We believe our approach is more practical since the SVD computations are made just once.

To summarize, in this work, we shifted focus a little into three areas: (i) classification into a large number of categories, (ii) evaluation of impact of training set size, and (iii) impact of some changes in the number of hidden units in the neural network. Overall, unfortunately, performance has not scaled very well. One possible reason for this came into light while the work was in progress: Dr. David Lewis announced that the original Reuters-22173 data had some anomalies in it. We decided to continue our experimentation with the same data for the reason that the new data that replaced the old collection was not pre-analyzed (requiring substantial initial time commitment) and also because we wanted to be able to compare the old and new results. We made some attempts (including personal communication) to find out the nature of the anomalies in the original Reuters data, but no information was available.

Acknowledgments

The author thanks the Air Force Office of Scientific Research (AFOSR) for the grant that made this work possible. Mr. Paul Robinson's assistance in this work is acknowledged as well.

References

1. Belkin, et al., 94: Belkin, N. J., P. Kantor, C. Cool, and R. Quatrain, Combining Evidence for Information Retrieval. In Harman, D. (Ed.), *The Second TREC Conference*, NIST Special Publication 500-215, pp. 35-44, 1994.
2. Cohen and Singer, 96: Cohen, W. and Y. Singer: Context-Sensitive Learning Methods for Text Categorization, *SIGIR-96*, pp. 307-315, 1996.
3. Dasigi, et al., 97: Dasigi, V., R. C. Mann, and V. Protopopescu, Multi-Sensor Text Classification Experiments - A Comparison, Oak Ridge National Laboratory Technical Memorandum ORNL/TM-13354, Oak Ridge, TN 37831, January, 1997.
4. Dasigi, 97: Dasigi, V., Information Fusion Experiments for Text Classification – An Experimental Comparison, *Report to AFOSR*, 1997.
5. Deerwester, et al., 90: Deerwester, S., S. Dumais, G. Furnas, T. Landauer, and R. Harshman, Indexing by Latent Semantic Analysis, *Journal of the American Society for Information Science*, 41(6), pp. 391-407, 1990.
6. Dumais, 96: Dumais, S., Combining Evidence for Effective Information Retrieval, *Working Notes of AAAI Spring Symposium of Machine Learning in Information Access*, pp. 26-30, March, 1996.
7. Forsythe, et al., 77: Forsythe, G., M. Malcolm, and C. Moler, *Computer Methods for Mathematical Computations* (Chapter 9: Least Squares and the Singular Value Decomposition), Prentice Hall, Englewood Cliffs, NJ, 1977.
8. Hull, et al., 96: Hull, D., J. Pedersen, and H. Schütze, Method Combination for Document Filtering, *SIGIR-96*, pp. 279-287, 1996.
9. Lewis, 92: Lewis, D., Representation and Learning in Information Retrieval, Ph.D. dissertation, University of Massachusetts, Amherst, February, 1992.
10. Qing, et al., 95: Qing, K., W. Caid, and C. Ren: Image / Text Automatic Indexing and Retrieval System using Context Vector Approach, *Proc. SPIE Digital Image Storage and Archiving Systems Conference*, pp. 372-379, 1995.
11. Schütze, et al., 95: Schütze, H., D. Hull, and J. Pedersen, A Comparison of Classifiers and Document Representations for the Routing Problem, *SIGIR-95*, pp. 229-237, 1995.
12. Towell, et al., 95: Towell, G., E. M. Voorhees, N. K. Gupta, and B. Johnson-Laird, Learning Collection Fusion Strategies for Information Retrieval, *Proc. Twelfth Annual Machine Learning Conference*, Lake Tahoe, July, 1995.
13. Uberbacher and Mural, 91: Uberbacher, E. and R. Mural, Locating Protein-Coding Regions in Human DNA Sequences by a Multiple Sensor-Neural Network Approach, *Proc. Natl. Acad. Sci. USA*, 88, pp. 11261-11265, 1991.
14. Xu, et al., 94: Xu, Y., R. Mural, M. Shah, and E. Uberbacher, Recognizing Exons in Genomic Sequence using GRAIL II, *Genetic Engineering, Principles and Methods*, Plenum Press, 15, June, 1994.

15. Zhou and Dapkus, 95: Zhou, J. and P. Dapkus, Automatic Suggestion of Significant Terms for a Pre-defined Topic, *Third ACL Workshop on Very Large Corpora*, MIT, Cambridge, June 1995.

Tables and Programs

Appendix A

MACRO / MICRO BREAKEVEN RESULTS FOR PRECISION AND RECALL

Experiment 1

Training Data *Number of documents: 4000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	10.18	0.04 – 0.05	96.42 – 97.74	27.54	0.07 – 0.08	99.27 – 99.48
32000	10.99	0.03 – 0.04	97.49 – 98.30	32.10	0.14 – 0.15	99.44 – 99.52
48000	14.22	0.02 – 0.03	95.36 – 97.22	34.03	0.14 – 0.05	99.45 – 99.53
64000	15.24	0.02 – 0.03	95.01 – 96.59	35.56	0.14 – 0.15	99.46 – 99.52
80000	13.59	0.03 – 0.04	97.22 – 97.95	36.04	0.11 – 0.12	99.46 – 99.52
96000	13.73	0.03 – 0.04	97.47 – 98.09	36.03	0.11 – 0.12	99.48 – 99.52
112000	13.41	0.03 – 0.04	97.54 – 98.15	36.49	0.12 – 0.13	99.50 – 99.55
128000	16.44	0.02 – 0.03	96.29 – 97.29	36.56	0.12 – 0.13	99.49 – 99.54
144000	14.45	0.03 – 0.04	97.52 – 98.13	36.55	0.12 – 0.13	99.48 – 99.52
160000	14.38	0.03 – 0.04	97.53 – 98.15	36.23	0.12 – 0.13	99.51 – 99.55
176000	16.26	0.02 – 0.03	96.19 – 97.34	36.82	0.13 – 0.14	99.52 – 99.56
192000	16.20	0.02 – 0.03	96.22 – 97.35	36.65	0.12 – 0.13	99.52 – 99.56

Testing Data *Number of documents: 18173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	5.00	0.04 – 0.05	96.56 – 97.78	26.57	0.07 – 0.08	99.10 – 99.33
32000	4.47	0.03 – 0.04	97.38 – 98.20	32.02	0.13 – 0.14	99.25 – 99.33

48000	0.00	0.04 – 1.02	97.93 – 99.49	0.00	0.14 – 1.02	99.31 – 99.49
64000	4.58	0.04 – 0.05	97.67 – 98.21	31.58	0.13 – 0.14	99.24 – 99.30
80000	0.00	0.04 – 1.04	97.81 – 99.49	0.00	0.12 – 1.04	99.32 – 99.49
96000	4.71	0.03 – 0.04	97.34 – 97.96	29.92	0.11 – 0.12	99.28 – 99.33
112000	4.57	0.03 – 0.04	97.40 – 98.00	30.44	0.11 – 0.12	99.25 – 99.31
128000	4.48	0.04 – 0.05	97.81 – 98.24	30.32	0.11 – 0.12	99.23 – 99.30
144000	4.43	0.03 – 0.04	97.36 – 97.97	30.65	0.12 – 0.13	99.28 – 99.33
160000	4.45	0.03 – 0.04	97.37 – 97.99	30.31	0.11 – 0.12	99.26 – 99.32
176000	4.45	0.04 – 0.05	97.86 – 98.28	30.67	0.12 – 0.13	99.26 – 99.32
192000	4.54	0.04 – 0.05	97.87 – 98.28	30.64	0.12 – 0.13	99.27 – 99.33

Experiment 2

Training Data *Number of documents: 8000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	5.69	0.04 – 0.05	96.86 – 97.47	22.54	0.14 – 0.15	99.34 – 99.43
32000	10.80	0.01 – 0.02	91.61 – 96.21	24.52	0.08 – 0.09	99.34 – 99.43
48000	8.78	0.02 – 0.03	95.40 – 97.39	23.33	0.06 – 0.07	99.17 – 99.37
64000	7.41	0.02 – 0.03	96.09 – 97.48	25.99	0.10 – 0.11	99.27 – 99.39
80000	8.86	0.03 – 0.04	97.11 – 98.06	27.89	0.11 – 0.12	99.35 – 99.45
96000	9.56	0.02 – 0.03	95.76 – 97.29	27.72	0.14 – 0.15	99.33 – 99.40
112000	9.45	0.02 – 0.03	95.95 – 97.31	27.82	0.13 – 0.14	99.37 – 99.46
128000	9.45	0.02 – 0.03	95.95 – 97.31	27.82	0.13 – 0.14	99.37 – 99.46
144000	10.30	0.02 – 0.03	95.62 – 97.16	28.04	0.11 – 0.12	99.35 – 99.45
160000	10.17	0.02 – 0.03	95.83 – 97.33	28.12	0.11 – 0.12	99.31 – 99.40
176000	9.92	0.02 – 0.03	95.83 – 97.33	28.01	0.11 – 0.12	99.30 – 99.39
192000	9.94	0.02 – 0.03	95.83 – 97.33	27.91	0.12 – 0.13	99.39 – 99.48

Testing Data *Number of documents: 14173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	4.86	0.05 – 0.06	97.51 – 97.86	27.22	0.13 – 0.14	99.18 – 99.29
32000	3.78	0.03 – 0.04	97.75 – 98.38	30.25	0.07 – 0.08	99.17 – 99.30
48000	4.22	0.03 – 0.04	97.42 – 98.31	28.38	0.06 – 0.07	99.12 – 99.32
64000	4.31	0.03 – 0.04	97.59 – 98.30	31.45	0.10 – 0.11	99.24 – 99.34
80000	4.62	0.03 – 0.04	97.21 – 98.10	33.40	0.10 – 0.11	99.24 – 99.33
96000	4.75	0.03 – 0.04	97.38 – 98.15	33.53	0.13 – 0.14	99.26 – 99.31
112000	4.79	0.03 – 0.04	97.43 – 98.14	33.27	0.11 – 0.12	99.25 – 99.33
128000	4.82	0.03 – 0.04	97.40 – 98.12	33.47	0.12 – 0.13	99.28 – 99.35
144000	4.70	0.03 – 0.04	97.25 – 98.03	33.28	0.10 – 0.11	99.23 – 99.32

160000	4.86	0.05 – 0.06	97.51 – 97.86	27.22	0.13 – 0.14	99.18 – 99.29
176000	4.83	0.03 – 0.04	97.41 – 98.16	33.50	0.11 – 0.12	99.29 – 99.36
192000	4.82	0.03 – 0.04	97.41 – 98.16	33.49	0.11 – 0.12	99.29 – 99.36

Experiment 3

Training Data *Number of documents: 12000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	6.45	0.03 – 0.04	95.97 – 97.34	22.22	0.07 – 0.08	99.19 – 99.35
32000	7.20	0.02 – 0.03	95.91 – 97.63	29.12	0.18 – 0.19	99.33 – 99.40
48000	5.66	0.03 – 0.04	96.84 – 97.84	29.54	0.12 – 0.13	99.25 – 99.35
64000	6.24	0.03 – 0.04	96.66 – 97.78	25.93	0.07 – 0.08	99.32 – 99.45
80000	9.36	0.01 – 0.02	91.52 – 95.87	29.69	0.12 – 0.13	99.28 – 99.39
96000	6.08	0.02 – 0.03	96.03 – 97.44	29.77	0.11 – 0.12	99.32 – 99.47
112000	7.37	0.02 – 0.03	96.11 – 97.51	29.64	0.12 – 0.13	99.25 – 99.36
128000	7.52	0.02 – 0.03	96.07 – 97.49	29.68	0.12 – 0.13	99.34 – 99.47
144000	6.87	0.02 – 0.03	95.72 – 97.28	29.55	0.11 – 0.12	99.29 – 99.43
160000	5.17	0.02 – 0.03	96.25 – 97.70	30.17	0.13 – 0.14	99.30 – 99.41
176000	5.17	0.02 – 0.03	96.21 – 97.68	30.06	0.13 – 0.14	99.31 – 99.42
192000	5.14	0.02 – 0.03	96.19 – 97.67	29.97	0.13 – 0.14	99.31 – 99.43

Testing Data *Number of documents: 10173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	3.36	0.04 – 0.05	97.33 – 98.27	23.74	0.07 – 0.08	99.12 – 99.30
32000	3.94	0.03 – 0.04	97.67 – 98.30	31.74	0.17 – 0.18	99.25 – 99.30
48000	6.22	0.02 – 0.03	95.43 – 96.93	31.71	0.11 – 0.12	99.22 – 99.31
64000	4.26	0.04 – 0.05	97.82 – 98.52	27.64	0.06 – 0.07	99.01 – 99.27
80000	4.53	0.03 – 0.04	97.65 – 98.42	32.62	0.12 – 0.13	99.26 – 99.36
96000	4.68	0.03 – 0.04	97.52 – 98.24	32.16	0.10 – 0.11	99.16 – 99.28
112000	4.49	0.03 – 0.04	97.59 – 98.28	32.55	0.12 – 0.13	99.21 – 99.33
128000	5.12	0.02 – 0.03	96.22 – 97.57	32.36	0.11 – 0.12	99.21 – 99.31
144000	5.18	0.03 – 0.04	97.38 – 98.15	31.83	0.11 – 0.12	99.26 – 99.39

160000	4.30	0.03 – 0.04	97.74 – 98.34	32.61	0.13 – 0.14	99.27 – 99.37
176000	4.32	0.03 – 0.04	97.73 – 98.33	32.54	0.13 – 0.14	99.28 – 99.38
192000	4.33	0.03 – 0.04	97.72 – 98.33	32.53	0.12 – 0.13	99.20 – 99.29

Experiment 4

Training Data *Number of documents: 16000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	3.96	0.03 – 0.04	97.35 – 98.06	25.09	0.15 – 0.16	99.28 – 99.30
32000	5.04	0.03 – 0.04	96.89 – 97.54	30.39	0.10 – 0.11	99.36 – 99.42
48000	4.88	0.03 – 0.04	97.39 – 98.20	30.57	0.08 – 0.09	99.44 – 99.52
64000	5.21	0.03 – 0.04	96.95 – 97.58	29.96	0.13 – 0.14	99.34 – 99.43
80000	5.96	0.02 – 0.03	97.60 – 98.28	31.91	0.12 – 0.13	99.31 – 99.38
96000	6.14	0.02 – 0.03	96.09 – 97.39	31.84	0.12 – 0.13	99.30 – 99.39
112000	5.35	0.02 – 0.03	95.98 – 97.27	31.96	0.13 – 0.14	99.24 – 99.30
128000	5.69	0.02 – 0.03	95.24 – 97.36	31.95	0.14 – 0.15	99.35 – 99.42
144000	5.70	0.02 – 0.03	96.09 – 97.52	32.11	0.12 – 0.13	99.32 – 99.39
160000	5.69	0.02 – 0.03	95.92 – 97.38	32.02	0.12 – 0.13	99.30 – 99.37
176000	5.68	0.02 – 0.03	95.93 – 97.39	31.95	0.12 – 0.13	99.30 – 99.36
192000	5.64	0.02 – 0.03	95.95 – 97.41	31.94	0.13 – 0.14	99.36 – 99.43

Testing Data *Number of documents: 6173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	3.77	0.04 – 0.05	98.04 – 98.46	23.75	0.13 – 0.14	99.16 – 99.19
32000	5.57	0.03 – 0.04	96.97 – 97.59	31.72	0.10 – 0.11	99.25 – 99.31
48000	5.00	0.03 – 0.04	97.41 – 98.20	31.47	0.07 – 0.08	99.10 – 99.26
64000	5.38	0.03 – 0.04	97.05 – 97.65	31.20	0.12 – 0.13	99.20 – 99.29
80000	5.67	0.02 – 0.03	96.30 – 97.66	32.94	0.12 – 0.13	99.26 – 99.34
96000	5.28	0.02 – 0.03	96.24 – 97.47	32.93	0.12 – 0.13	99.26 – 99.34
112000	5.42	0.02 – 0.03	96.14 – 97.36	32.82	0.13 – 0.14	99.25 – 99.32
128000	5.31	0.02 – 0.03	96.11 – 97.43	32.85	0.13 – 0.14	99.25 – 99.31
144000	5.53	0.02 – 0.03	96.25 – 97.59	33.04	0.11 – 0.12	99.21 – 99.27

160000	5.64	0.02 – 0.03	96.08 – 97.46	32.92	0.12 – 0.13	99.26 – 99.33
176000	5.66	0.02 – 0.03	96.10 – 97.47	32.90	0.12 – 0.13	99.25 – 99.32
192000	5.66	0.02 – 0.03	96.11 – 97.49	32.93	0.12 – 0.13	99.25 – 99.31

Experiment 5

Training Data *Number of documents: 20000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	4.19	0.02 – 0.03	96.61 – 97.16	25.72	0.09 – 0.10	99.24 – 99.33
32000	4.78	0.03 – 0.04	96.97 – 97.67	29.97	0.14 – 0.15	99.32 – 99.42
48000	5.38	0.03 – 0.04	96.67 – 97.49	27.97	0.09 – 0.10	99.18 – 99.39
64000	4.11	0.02 – 0.03	96.10 – 97.13	29.39	0.18 – 0.19	99.26 – 99.34
80000	5.31	0.02 – 0.03	96.30 – 97.54	30.72	0.13 – 0.14	99.29 – 99.41
96000	5.99	0.02 – 0.03	95.77 – 97.31	31.01	0.12 – 0.13	99.25 – 99.42
112000	5.79	0.02 – 0.03	96.13 – 97.40	30.98	0.14 – 0.15	99.30 – 99.45
128000	6.54	0.02 – 0.03	95.81 – 97.31	31.03	0.14 – 0.15	99.31 – 99.48
144000	5.50	0.02 – 0.03	96.15 – 97.60	31.00	0.13 – 0.14	99.25 – 99.38
160000	8.25	0.01 – 0.02	92.10 – 95.57	30.95	0.15 – 0.16	99.29 – 99.39
176000	8.22	0.01 – 0.02	92.06 – 95.59	30.96	0.15 – 0.16	99.31 – 99.44
192000	8.19	0.01 – 0.02	92.02 – 95.60	30.82	0.14 – 0.15	99.24 – 99.34

Testing Data *Number of documents: 2173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	6.73	0.01 – 0.02	95.09 – 96.49	30.89	0.08 – 0.09	99.06 – 99.21
32000	5.19	0.03 – 0.04	97.16 – 97.79	31.25	0.13 – 0.14	99.15 – 99.25
48000	5.55	0.04 – 0.05	97.62 – 98.01	30.19	0.09 – 0.10	99.11 – 99.31
64000	4.90	0.03 – 0.04	97.21 – 97.73	31.06	0.18 – 0.19	99.19 – 99.27
80000	5.82	0.02 – 0.03	96.54 – 97.61	31.20	0.12 – 0.13	99.11 – 99.22
96000	6.11	0.02 – 0.03	96.02 – 97.45	31.16	0.12 – 0.13	99.17 – 99.33
112000	6.53	0.02 – 0.03	96.36 – 97.50	31.91	0.13 – 0.14	99.15 – 99.23
128000	6.51	0.02 – 0.03	96.05 – 97.43	32.44	0.13 – 0.14	99.15 – 99.24
144000	8.59	0.01 – 0.02	93.44 – 96.39	32.17	0.13 – 0.14	99.19 – 99.31
160000	6.21	0.03 – 0.04	97.21 – 97.97	32.22	0.14 – 0.15	99.14 – 99.22

176000	6.76	0.02 – 0.03	95.88 – 97.22	32.23	0.14 – 0.15	99.16 – 99.25
192000	6.70	0.02 – 0.03	95.89 – 97.23	32.12	0.14 – 0.15	99.17 – 99.28

NEURAL NETWORK TRAINING EXPERIMENTS FOR LARGE SCALE CLASSIFICATION

Network with 135 inputs, 135 outputs and 15 hidden nodes.

Experiment 1

Training Data *Number of documents: 4000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	10.81	0.02 – 0.03	96.08 – 96.95	25.09	0.14 – 0.15	99.38 – 99.43
32000	13.64	0.04 – 0.05	97.13 – 97.94	31.92	0.10 – 0.11	99.41 – 99.49
48000	18.47	0.02 – 0.03	95.75 – 96.99	34.81	0.11 – 0.12	99.45 – 99.51
64000	19.75	0.02 – 0.03	95.99 – 97.17	36.26	0.11 – 0.12	99.49 – 99.54
80000	18.77	0.03 – 0.04	97.41 – 98.03	38.42	0.12 – 0.13	99.52 – 99.57
96000	19.06	0.03 – 0.04	97.29 – 97.94	38.76	0.12 – 0.13	99.53 – 99.56
112000	18.91	0.03 – 0.04	97.37 – 98.03	38.76	0.11 – 0.12	99.48 – 99.54
128000	18.85	0.03 – 0.04	97.50 – 98.12	39.14	0.12 – 0.13	99.53 – 99.57
144000	19.06	0.03 – 0.04	97.43 – 98.07	38.69	0.13 – 0.14	99.52 – 99.56
160000	19.19	0.03 – 0.04	97.31 – 97.96	38.37	0.11 – 0.12	99.51 – 99.55
176000	19.20	0.03 – 0.04	97.33 – 97.97	38.76	0.11 – 0.12	99.51 – 99.55
192000	19.24	0.03 – 0.04	97.34 – 97.98	38.86	0.11 – 0.12	99.50 – 99.54

Testing Data *Number of documents: 18173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	4.42	0.03 – 0.04	97.07 – 97.59	27.36	0.13 – 0.14	99.21 – 99.28
32000	5.22	0.04 – 0.05	97.15 – 97.87	27.08	0.10 – 0.11	99.23 – 99.31
48000	6.20	0.03 – 0.04	96.86 – 97.56	28.79	0.11 – 0.12	99.32 – 99.37
64000	6.47	0.02 – 0.03	95.85 – 97.04	29.86	0.10 – 0.14	99.21 – 99.30
80000	5.93	0.03 – 0.04	97.25 – 97.89	30.74	0.11 – 0.12	99.26 – 99.33
96000	6.02	0.03 – 0.04	97.16 – 97.80	29.93	0.11 – 0.12	99.26 – 99.32
112000	5.99	0.03 – 0.04	97.20 – 97.86	30.34	0.11 – 0.12	99.27 – 99.33
128000	5.83	0.03 – 0.04	97.31 – 97.34	30.69	0.11 – 0.12	99.26 – 99.32
144000	6.02	0.03 – 0.04	97.26 – 97.87	30.58	0.12 – 0.13	99.26 – 99.31
160000	5.64	0.03 – 0.04	97.15 – 97.80	29.44	0.10 – 0.11	99.22 – 99.29
176000	5.63	0.03 – 0.04	97.17 – 97.82	29.76	0.10 – 0.11	99.21 – 99.28
192000	5.63	0.03 – 0.04	97.19 – 97.83	29.92	0.11 – 0.12	99.28 – 99.33

Network with 135 inputs, 135 outputs and 15 hidden nodes.

Experiment 5

Training Data *Number of documents: 20000*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	0.00	0.04 – 0.99	96.60 – 99.52	00.00	0.17 – 0.99	99.33 – 99.52
32000	5.95	0.02 – 0.03	96.41 – 97.61	30.84	0.12 – 0.13	99.32 – 99.40
48000	3.22	0.03 – 0.04	98.16 – 98.50	30.86	0.12 – 0.13	99.23 – 99.34
64000	9.15	0.01 – 0.02	94.54 – 97.25	30.24	0.15 – 0.16	99.33 – 99.38
80000	9.35	0.01 – 0.02	96.47 – 97.52	31.73	0.13 – 0.14	99.29 – 99.38
96000	6.56	0.01 – 0.02	94.52 – 96.68	32.05	0.14 – 0.15	99.30 – 99.39
112000	6.46	0.02 – 0.03	96.28 – 97.31	32.15	0.12 – 0.13	99.32 – 99.42
128000	6.16	0.02 – 0.03	96.56 – 97.55	32.18	0.14 – 0.15	99.29 – 99.36
144000	6.70	0.02 – 0.03	96.31 – 98.10	32.22	0.14 – 0.15	99.33 – 99.42
160000	7.93	0.01 – 0.02	96.71 – 97.68	32.22	0.13 – 0.14	99.33 – 99.42
176000	7.90	0.01 – 0.02	94.39 – 96.71	32.29	0.13 – 0.14	99.33 – 99.41
192000	7.85	0.01 – 0.02	94.37 – 96.70	32.31	0.13 – 0.14	99.32 – 99.41

Testing Data *Number of documents: 2173*

Learning Iterations	MACRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)	MICRO Breakeven (%)	Threshold (Occurred Between)	Correctness (%)
16000	6.73	0.05 – 0.06	96.83 – 97.23	27.72	0.15 – 0.16	99.12 – 99.23
32000	5.42	0.02 – 0.03	97.06 – 98.15	32.04	0.12 – 0.13	99.16 – 99.26
48000	7.00	0.02 – 0.03	96.39 – 97.61	31.19	0.11 – 0.12	99.17 – 99.24
64000	5.64	0.02 – 0.03	97.32 – 98.13	31.80	0.13 – 0.14	99.16 – 99.21
80000	7.21	0.02 – 0.03	96.69 – 97.62	32.57	0.12 – 0.13	99.16 – 99.22
96000	6.76	0.02 – 0.03	96.85 – 97.70	32.47	0.13 – 0.14	99.17 – 99.24
112000	6.66	0.03 – 0.04	97.43 – 97.99	32.32	0.11 – 0.12	99.16 – 99.25
128000	6.92	0.02 – 0.03	96.72 – 97.62	32.75	0.13 – 0.14	99.17 – 99.23
144000	7.21	0.02 – 0.03	96.53 – 97.50	32.90	0.13 – 0.14	99.20 – 99.27
160000	7.09	0.02 – 0.03	96.91 – 97.74	32.53	0.12 – 0.13	99.18 – 99.26
176000	7.09	0.02 – 0.03	96.90 – 97.22	32.55	0.12 – 0.13	99.18 – 99.25
192000	7.08	0.02 – 0.03	96.90 – 97.73	32.56	0.12 – 0.13	99.18 – 99.25

Appendix B

Neural Networks Training and Testing Data Results Example

SMN32K20.txt - Output for 32000 iteration for the 20000 training set size.

Minimum: -0.074 Maximum: 0.980

Thres	MacPre	MacRec	MicPre	MicRec	MacCor	MicCor
-0.066	0.48	100.00	0.48	100.00	0.48	0.48
-0.056	0.48	100.00	0.48	100.00	0.48	0.48
-0.046	0.48	99.99	0.48	99.96	0.48	0.48
-0.036	0.48	99.98	0.48	99.91	0.50	0.50
-0.026	0.48	99.91	0.48	99.56	0.56	0.56
-0.016	0.49	99.57	0.47	98.42	0.93	0.93
-0.006	0.50	96.89	0.47	95.10	3.66	3.66
0.004	1.07	27.75	2.23	77.95	83.48	83.48
0.014	3.45	11.45	4.70	63.89	93.63	93.63
0.024	5.56	5.60	8.72	55.37	97.01	97.01
0.034	3.18	3.23	13.15	50.78	98.16	98.16
0.044	5.15	2.59	15.58	47.99	98.50	98.50
0.054	5.71	2.25	17.15	45.57	98.68	98.68
0.064	10.21	2.04	18.68	43.62	98.82	98.82
0.074	10.67	1.87	19.98	41.98	98.92	98.92
0.084	15.33	1.73	21.16	40.60	98.99	98.99
0.094	16.65	1.65	22.10	39.45	99.04	99.04
0.104	12.78	1.55	23.09	38.23	99.09	99.09
0.114	15.60	1.45	24.41	36.75	99.15	99.15
0.124	15.73	1.32	26.44	34.45	99.23	99.23
0.134	20.20	1.14	31.01	30.74	99.34	99.34
0.144	17.12	0.90	38.26	25.96	99.44	99.44
0.154	21.28	0.75	47.69	22.94	99.51	99.51
0.164	21.61	0.68	56.02	21.56	99.54	99.54
0.174	19.16	0.64	61.38	20.85	99.56	99.56

0.184	19.12	0.63	65.34	20.59	99.57	99.57
0.194	18.05	0.62	67.72	20.46	99.57	99.57
0.204	20.01	0.62	69.58	20.45	99.58	99.58
0.214	29.14	0.62	70.61	20.38	99.58	99.58
0.224	23.84	0.62	71.44	20.34	99.58	99.58
0.234	36.14	0.62	72.23	20.28	99.58	99.58
0.244	36.51	0.62	73.01	20.24	99.58	99.58
0.254	73.83	0.61	73.83	20.17	99.58	99.58
0.264	74.39	0.61	74.39	20.12	99.58	99.58
0.274	75.07	0.61	75.07	20.07	99.59	99.59
0.284	75.53	0.61	75.53	20.01	99.59	99.59
0.294	76.04	0.61	76.04	19.97	99.59	99.59
0.304	76.62	0.61	76.62	19.91	99.59	99.59
0.314	77.26	0.61	77.26	19.88	99.59	99.59
0.324	77.57	0.60	77.57	19.86	99.59	99.59
0.334	77.85	0.60	77.85	19.80	99.59	99.59
0.344	78.17	0.60	78.17	19.76	99.59	99.59
0.354	78.44	0.60	78.44	19.74	99.59	99.59
0.364	78.99	0.60	78.99	19.70	99.59	99.59
0.374	79.50	0.60	79.50	19.70	99.59	99.59
0.384	79.78	0.60	79.78	19.67	99.59	99.59
0.394	80.04	0.60	80.04	19.65	99.59	99.59
0.404	80.49	0.60	80.49	19.62	99.59	99.59
0.414	80.73	0.60	80.73	19.59	99.59	99.59
0.424	80.95	0.60	80.95	19.55	99.59	99.59
0.434	81.29	0.60	81.29	19.54	99.59	99.59
0.444	81.85	0.59	81.85	19.53	99.59	99.59
0.454	82.15	0.59	82.15	19.49	99.59	99.59
0.464	82.57	0.59	82.57	19.48	99.59	99.59
0.474	82.86	0.59	82.86	19.46	99.59	99.59
0.484	83.09	0.59	83.09	19.44	99.60	99.60
0.494	83.43	0.59	83.43	19.42	99.60	99.60
0.504	83.86	0.59	83.86	19.40	99.60	99.60
0.514	84.07	0.59	84.07	19.37	99.60	99.60
0.524	84.56	0.59	84.56	19.35	99.60	99.60
0.534	84.87	0.59	84.87	19.34	99.60	99.60

0.544	85.19	0.59	85.19	19.29	99.60	99.60
0.554	85.37	0.59	85.37	19.26	99.60	99.60
0.564	85.65	0.59	85.65	19.23	99.60	99.60
0.574	86.04	0.58	86.04	19.20	99.60	99.60
0.584	86.28	0.58	86.28	19.11	99.60	99.60
0.594	86.48	0.58	86.48	19.09	99.60	99.60
0.604	86.83	0.58	86.83	19.06	99.60	99.60
0.614	87.35	0.58	87.35	18.95	99.60	99.60
0.624	87.59	0.58	87.59	18.88	99.60	99.60
0.634	87.89	0.57	87.89	18.73	99.60	99.60
0.644	88.39	0.56	88.39	18.54	99.60	99.60
0.654	88.81	0.56	88.81	18.34	99.60	99.60
0.664	89.22	0.55	89.22	18.16	99.60	99.60
0.674	89.65	0.55	89.65	17.95	99.60	99.60
0.684	90.01	0.54	90.01	17.69	99.60	99.60
0.694	90.54	0.53	90.54	17.45	99.60	99.60
0.704	90.83	0.52	90.83	17.15	99.59	99.59
0.714	91.25	0.51	91.25	16.77	99.59	99.59
0.724	91.84	0.50	91.84	16.36	99.59	99.59
0.734	92.07	0.49	92.07	16.07	99.59	99.59
0.744	92.39	0.48	92.39	15.78	99.59	99.59
0.754	92.71	0.47	92.71	15.44	99.59	99.59
0.764	92.73	0.46	92.73	15.19	99.59	99.59
0.774	92.89	0.46	92.89	15.04	99.59	99.59
0.784	93.26	0.45	93.26	14.87	99.59	99.59
0.794	93.47	0.45	93.47	14.72	99.59	99.59
0.804	93.80	0.45	93.80	14.62	99.59	99.59
0.814	94.18	0.44	94.18	14.50	99.59	99.59
0.824	94.51	0.44	94.51	14.38	99.59	99.59
0.834	94.71	0.43	94.71	14.25	99.59	99.59
0.844	94.86	0.43	94.86	14.13	99.58	99.58
0.854	95.21	0.43	95.21	13.98	99.58	99.58
0.864	95.51	0.42	95.51	13.81	99.58	99.58
0.874	95.61	0.41	95.61	13.46	99.58	99.58
0.884	95.91	0.40	95.91	13.06	99.58	99.58
0.894	95.95	0.38	95.95	12.44	99.58	99.58

0.904	96.17	0.35	96.17	11.63	99.57	99.57
0.914	96.09	0.31	96.09	10.26	99.57	99.57
0.924	96.00	0.23	96.00	7.60	99.56	99.56
0.934	94.61	0.11	94.61	3.66	99.54	99.54
0.944	95.22	0.05	95.22	1.54	99.53	99.53
0.954	97.40	0.02	97.40	0.58	99.52	99.52
0.964	92.00	0.01	92.00	0.18	99.52	99.52
0.974	83.33	0.00	83.33	0.04	99.52	99.52

MACRO BREAKEVEN	=	3.22	occurred between:	0.03	0.04
MICRO BREAKEVEN	=	30.86	occurred between:	0.12	0.13

Appendix C

Statistical Information about raw Reuters-22173 data before creating input for neural network experiments

Number of documents assigned to category 1 = 4075

Number of documents assigned to category 3 = 227

Number of documents assigned to category 4 = 827

Number of documents assigned to category 6 = 54

Number of documents assigned to category 8 = 530

Number of documents assigned to category 9 = 169

Number of documents assigned to category 11 = 44

Number of documents assigned to category 12 = 3

Number of documents assigned to category 13 = 197

Number of documents assigned to category 14 = 140

Number of documents assigned to category 27 = 2506

Number of documents assigned to category 36 = 37

Number of documents assigned to category 48 = 78

Number of documents assigned to category 51 = 308

Number of documents assigned to category 52 = 70

Number of documents assigned to category 55 = 316

Number of documents assigned to category 56 = 646

Number of documents assigned to category 57 = 151

Number of documents assigned to category 60 = 191

Number of documents assigned to category 62 = 138

Number of documents assigned to category 65 = 6

Number of documents assigned to category 68 = 37

Number of documents assigned to category 69 = 52

Number of documents assigned to category 81 = 27

Number of documents assigned to category 82 = 67

Number of documents assigned to category 84 = 119

Number of documents assigned to category 86 = 55

Number of documents assigned to category 87 = 255

Number of documents assigned to category 88 = 34

Number of documents assigned to category 104 = 114

Number of documents assigned to category 107 = 28

Number of documents assigned to category 108 = 118

Number of documents assigned to category 112 = 36

Number of documents assigned to category 113 = 124

Number of documents assigned to category 115 = 15

Number of documents assigned to category 116 = 80

Number of documents assigned to category 119 = 18

Number of documents assigned to category 120 = 65

Number of documents assigned to category 122 = 22

Number of documents assigned to category 123 = 1

Number of documents assigned to category 124 = 3

Number of documents assigned to category 125 = 1

Number of documents assigned to category 130 = 33

Number of documents assigned to category 135 = 37

Number of documents assigned to category 137 = 79

Number of documents assigned to category 149 = 2

Number of documents assigned to category 150 = 10

Number of documents assigned to category 155 = 77

Number of documents assigned to category 157 = 639

Number of documents assigned to category 159 = 133

Number of documents assigned to category 160 = 29

Number of documents assigned to category 164 = 3

Number of documents assigned to category 165 = 1

Number of documents assigned to category 169 = 569

Number of documents assigned to category 172 = 71

Number of documents assigned to category 173 = 78

Number of documents assigned to category 177 = 32

Number of documents assigned to category 178 = 19

Number of documents assigned to category 179 = 27

Number of documents assigned to category 186 = 66

Number of documents assigned to category 187 = 1

Number of documents assigned to category 190 = 13

Number of documents assigned to category 192 = 63

Number of documents assigned to category 197 = 44

Number of documents assigned to category 202 = 15

Number of documents assigned to category 203 = 3

Number of documents assigned to category 205 = 84

Number of documents assigned to category 206 = 192

Number of documents assigned to category 207 = 17

Number of documents assigned to category 221 = 29

Number of documents assigned to category 239 = 42

Number of documents assigned to category 242 = 25

Number of documents assigned to category 243 = 27

Number of documents assigned to category 247 = 12

Number of documents assigned to category 254 = 2

Number of documents assigned to category 255 = 4

Number of documents assigned to category 267 = 4

Number of documents assigned to category 280 = 6

Number of documents assigned to category 282 = 6

Number of documents assigned to category 312 = 7

Number of documents assigned to category 314 = 1

Number of documents assigned to category 316 = 3

Number of documents assigned to category 318 = 16

Number of documents assigned to category 335 = 9

Number of documents assigned to category 339 = 3

Number of documents assigned to category 341 = 3

Number of documents assigned to category 342 = 3

Number of documents assigned to category 343 = 8

Number of documents assigned to category 344 = 9

Number of documents assigned to category 345 = 7

Number of documents assigned to category 347 = 12

Number of documents assigned to category 353 = 3

Number of documents assigned to category 354 = 1

Number of documents assigned to category 355 = 4

Number of documents assigned to category 358 = 21

Number of documents assigned to category 360 = 2

Number of documents assigned to category 367 = 1

Number of documents assigned to category 370 = 2

Number of documents assigned to category 376 = 3

Number of documents assigned to category 386 = 7

Number of documents assigned to category 432 = 2

Number of documents assigned to category 455 = 1

Number of documents assigned to category 461 = 7

Number of documents assigned to category 477 = 4

Number of documents assigned to category 478 = 1

Number of documents assigned to category 503 = 8

Number of documents assigned to category 505 = 3

Number of documents assigned to category 507 = 3

Number of documents assigned to category 515 = 4

Number of documents assigned to category 516 = 2

Number of documents assigned to category 536 = 4

Number of documents assigned to category 539 = 1

Number of documents assigned to category 586 = 1

Number of documents assigned to category 592 = 3

Number of documents assigned to category 598 = 1

Number of documents assigned to category 619 = 2

Number of documents assigned to category 661 = 1

Number of documents assigned to category 668 = 1

Number of documents assigned to category 673 = 0

Number of documents assigned to category 674 = 0

Number of documents assigned to category 675 = 0

Number of documents assigned to category 676 = 0

Number of documents assigned to category 677 = 0

Number of documents assigned to category 678 = 0

Number of documents assigned to category 679 = 0

Number of documents assigned to category 680 = 0

Number of documents assigned to category 681 = 0

Number of documents assigned to category 682 = 0

Number of documents assigned to category 683 = 0

Number of documents assigned to category 684 = 0

Number of documents assigned to category 685 = 0

Number of documents assigned to category 686 = 0

Number of documents assigned to category 687 = 0

Number of documents assigned to category 688 = 0

Number of documents assigned to category 689 = 0

Total number of documents with category assignments = 14652

Number of documents with topic(s) assigned = 11664

Number of documents with no assignments at all = 10509

Percentage of documents categorized = 66.08

Appendix D

List of categories

Money/Foreign Exchange (MONEY-FX)
Shipping (SHIP)
Interest Rates (INTEREST)

****Economic Indicator Codes (16)**

Balance of Payments (BOP)
Trade (TRADE)
Consumer Price Index (CPI)
Wholesale Price Index (WPI)
Unemployment (JOBS)
Industrial Production Index (IPI)
Capacity Utilization (CPU)
Gross National/Domestic Product (GNP)
Money Supply (MONEY-SUPPLY)
Reserves (RESERVES)
Leading Economic Indicators (LEI)
Housing Starts (HOUSING)
Personal Income (INCOME)
Inventories (INVENTORIES)
Installment Debt/Consumer Credit (INSTAL-DEBT)
Retail Sales (RETAIL)

****Currency Codes (27)**

U.S. Dollar (DLR)
Australian Dollar (AUSTDLR)
Hong Kong Dollar (HK)
Singapore Dollar (SINGDLR)
New Zealand Dollar (NZDLR)
Canadian Dollar (CAN)
Sterling (STG)
D-Mark (DMK)
Japanese Yen (YEN)
Swiss Franc (SFR)
French Franc (FFR)
Belgian Franc (BFR)

Netherlands Guilder/Florin (DFL)

Italian Lira (LIT)

Danish Krone/Crown (DKR)

Norwegian Krone/Crown (NKR)

Swedish Krona/Crown (SKR)

Mexican Peso (MEXPESO)

Brazilian Cruzado (CRUZADO)

Argentine Austral (AUSTRAL)

Saudi Arabian Riyal (SAUDRIYAL)

South African Rand (RAND)

Indonesian Rupiah (RUPIAH)

Malaysian Ringitt (RINGGIT)

Portuguese Escudo (ESCUDO)

Spanish Peseta (PESETA)

Greek Drachma (DRACHMA)

<u>**Corporate Codes (2)</u>

Mergers/Acquisitions (ACQ)

Earnings and Earnings Forecasts (EARN)

<u>**Commodity Codes (78)</u>

ALUM

BARLEY

CARCASS

CASTOR-MEAL

CASTOR-OIL

CASTORSEED

CITRUSPULP

COCOA

COCONUT-OIL

COCONUT

COFFEE

COPPER

COPRA-CAKE

CORN-OIL

CORN

CORNGLUTENFEED

COTTON

COTTON-MEAL

COTTON-OIL

COTTONSEED
F-CATTLE
FISHMEAL
FLAXSEED
GOLD
GRAIN
GROUNDNUT
GROUNDNUT-MEAL
GROUNDNUT-OIL
IRON-STEEL
LEAD
LIN-MEAL
LIN-OIL
LINSEED
LIVESTOCK
L-CATTLE
HOG
LUMBER
LUPIN
MEAL-FEED
NICKEL
OAT
OILSEED
ORANGE
PALLADIUM
PALM-MEAL
PALM-OIL
PALMKERNEL
PLATINUM
PLYWOOD
PORK-BELLY
POTATO
RAPE-MEAL
RAPE-OIL
RAPESEED
RED-BEAN
RICE
RUBBER
RYE
SILK

SILVER

SORGHUM

SOY-MEAL

SOY-OIL

SOYBEAN

STRATEGIC-METAL

SUGAR

SUN-MEAL

SUN-OIL

SUNSEED

TAPIOCA

TEA

TIN

TUNG-OIL

TUNG

VEG-OIL

WHEAT

WOOL

ZINC

Energy Codes (9)

Crude Oil (CRUDE)

Heating Oil/Gas Oil (HEAT)

Fuel Oil (FUEL)

Gasoline (GAS)

Natural Gas (NAT-GAS)

Petro-Chemicals (PET-CHEM)

Propane (PROPANE)

Jet and Kerosene (JET)

Naphtha (NAPHTHA)

THE INTERACTIVE LEARNING WALL: A PC-BASED, DEPLOYABLE DATA WALL FOR USE IN A COLLEGE CLASSROOM

Richard R. Eckert
Associate Professor
Department of Computer Science
Watson School of Engineering

Binghamton University
Vestal Parkway East
Binghamton, NY 13902

Final Report for:
Summer Faculty Research Program
Rome Laboratory

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Rome Laboratory

December. 1998

THE INTERACTIVE LEARNING WALL: A PC-BASED, DEPLOYABLE DATA WALL FOR USE IN A COLLEGE CLASSROOM

Richard R. Eckert
Associate Professor
Department of Computer Science
Binghamton University

Abstract

The primary focus of this project was the development of a Windows/PC-based Interactive Learning Wall which creates a virtual blackboard that can be controlled at a distance by a classroom instructor. The most important component of the system is a wireless mouse emulator that is implemented using a software-controlled standard red laser pointer. With this system the instructor is no longer tethered to the computer console. A video camera looks at the projected screen and feeds its output to the computer. Our programs detect and interpret the location of the laser beam, map it to screen coordinates, and move the system mouse cursor accordingly, thus simulating system mouse actions. Using a unique algorithm to correct for the geometry between the camera and the screen, we can accurately point, click, drag, and perform other mouse manipulation actions with the laser pointer. A secondary component of the Interactive Learning Wall is a wireless LAN system that permits communication between student laptop computers and the instructor's computer. Using this system and programs we developed, students can make requests from their laptops to take control of the main computer's pointer device and keyboard. This allows them to interact directly with the instructor's computer in order to perform tasks like pointing to specific parts of the presentation for clarification of doubts and sharing of information. A third aspect of the project was an evaluation of off-the-shelf voice command recognition software which would enable the instructor to control his presentation with voice commands. A final aspect of this project was to provide an environment in which both graduate and undergraduate students could work as a team on a real-world human-computer interface problem.

THE INTERACTIVE LEARNING WALL: A PC-BASED, DEPLOYABLE DATA WALL FOR USE IN A COLLEGE CLASSROOM

Richard R. Eckert

Introduction

Teaching and learning involve the effective exchange of information between an expert and a group of novices. This kind of activity permits the novice to understand and perform new tasks. Although traditionally we think of teaching and learning as occurring in a school setting, the concepts can be applied to any realms of human endeavor--including the military. In that realm, the effective exchange of information is particularly important.

Some of the most effective teaching takes place in one-on-one or small group settings. This kind of environment provides for individual attention and an easy, non-intimidating rapport between student and teacher that is conducive to learning. As the material to be learned becomes more complex and voluminous, the number of learners (novices) becomes much greater than the number of teachers (experts), so that the small-group setting becomes impractical. To facilitate the presentation to larger student groups, the teacher must have access to some sort of display device to post lessons and/or notes. Artifacts such as blackboards, whiteboards, overhead projection systems, and computerized projection systems have become very prevalent. Unfortunately, in large-group settings that use this kind of equipment, interaction between the teacher and individual students is kept to a minimum at least partly because of the fact that the teacher is tethered to the display system.

Computers have added another dimension to today's classroom by greatly extending the type of data that can be experienced by students and teachers. Multimedia technology has provided interactive applications, lessons, and presentations that capture the audience. Slide show software such as Microsoft PowerPoint™ has been used in professional and academic presentations and provides stunningly attractive artistic layouts. Complex three-dimensional graphics can easily be visualized and manipulated. Teleconferencing can bring experts into the classroom and reduce the immense cost normally associated with guest speakers. The World Wide Web has provided a window into the universe--an infinite source of learning materials--that is now becoming accessible to classrooms. Unfortunately many faculty members do not feel comfortable with the vast array of equipment they are confronted with, and, hence, do not use it willingly. Making use of these devices more natural, intuitive, and interactive has thus become very important.

The Interactive Data Wall

The Advanced Displays and Intelligent Interfaces group at the Rome, NY Air Force Research Laboratory has created a sophisticated and expensive presentation system that involves many of the above-mentioned components. Its "Interactive DataWall" consists of an ultra-high-resolution wall display driven by an SGI ONYX computer with various types of wireless input devices shared among multiple users. The devices include: speech interaction, camera-tracked laser pointers, and flight sticks. The purpose of the DataWall is to display large amounts of information in a contiguous area and to provide multiple users with an intuitive and unencumbered means of accessing and manipulating the data. [1]

In this project, two graduate students, six undergraduates and the project director were able to develop an Interactive Learning Wall. This is a scaled-down, PC/Windows-based implementation of the Rome Laboratory facility designed for use in a civilian environment. More precisely, the Learning Wall adapts the Air Force technology to benefit a modern classroom. It should also be pointed out that its successful implementation can be of benefit to the Air Force, since with an inexpensive PC, wireless networking capability, a readily-available RGB projection system, an inexpensive video camera and a laser pointer, decision makers can effectively exchange information and have access to the world.

The Interactive Learning Wall

Figure 1 shows the interactive learning wall setup.

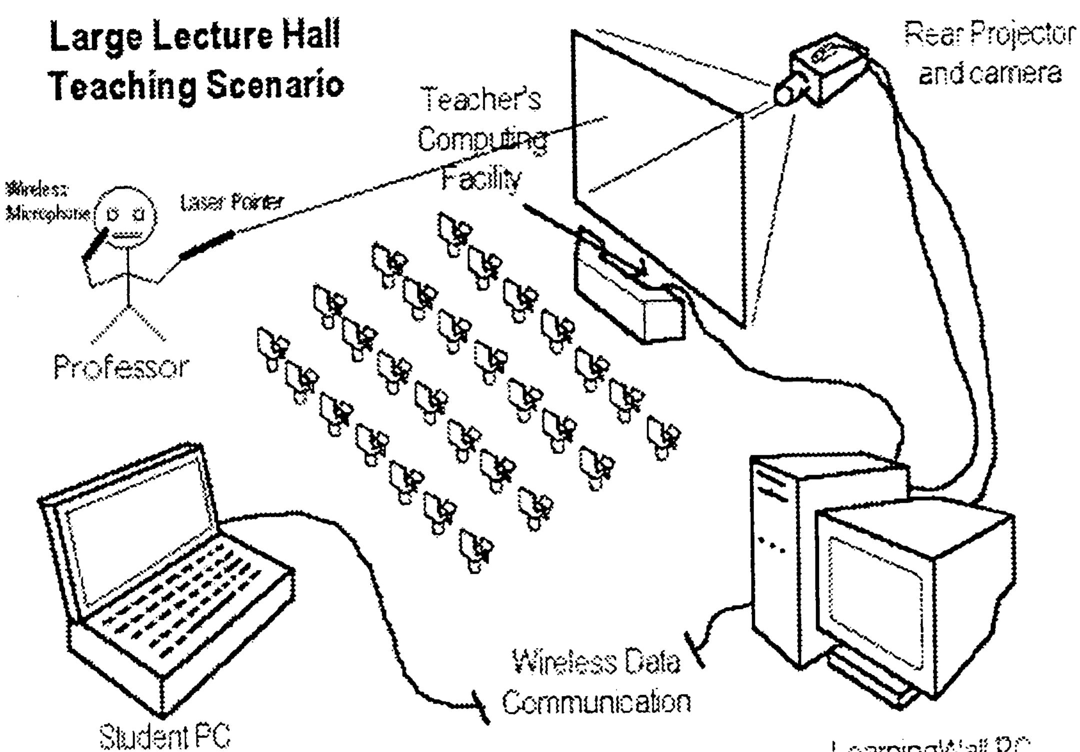


Figure 1. Rear Screen Learning Wall Setup.

Although a rear-screen configuration is shown, the system can also be used with the projector and camera in front of the projection screen. It is important, however, in a front-screen setup that the operator not shadow the image being displayed by the projector and viewed by the camera. The projector projects whatever content the professor is displaying on the screen. This could be a PowerPoint™ slide presentation or any other visual material from the computer. The professor uses the laser pointer to point at and interact with the material on the screen. The interaction is very reminiscent of that obtained with a standard mouse. The camera detects the bright laser spot on the screen, and our software computes the corresponding location on the screen and moves a "mouse cursor" icon accordingly.

In our implementation, the lower right-hand corner of the screen displays a white box that shows the current mode of operation. Upon startup it says LEFT. This means that if the user "clicks" with the laser pointer, it generates the same effect that hitting the left mouse button would generate. The interface is similar to that used with touch pads on some laptop computers. The first time the user directs the laser pointer at the screen it is treated as a location identifier. In other words the user needs to accurately point at the location where he or she would like to push the button. To generate a single click the user releases the pointer, and within one second in close proximity to the previous location, presses the pen again. To generate a double click this process is repeated. Although this may seem like an odd implementation, to achieve accuracy at a distance it is essential to take into account the jitter of a user's hand. In our implementation, the first time the laser point is seen is used as an "aim" and the second is a "fire."

The other modes of operation are RIGHT and DRAW, and they are selected by turning off the laser pointer within the white box in the lower right-hand corner of the screen. In RIGHT mode the program generates right mouse button clicks. In DRAW mode a color bar appears to the right of the screen and displays four colors; yellow, red, blue and green. The active color is highlighted by a white border. To draw the user only needs to click once and the program will continue to draw until the laser pointer is released. In order to switch colors the user releases the laser pointer over the desired color.

The computer

Since real-time digital video processing is required, a fairly well-endowed computer with a video capture card is necessary to allow implementation of the Learning Wall. The system also has to simultaneously handle user, system, and application events. We used a computer with two Intel Pentium II 300 MHz processors, 128 megabytes of SDRAM, an IBM 8 gigabyte ultra wide SCSI hard drive, an 8 megabyte ATI Expert@Work 2x AGP video card, and a Computer Eyes/PCI video capture card. This was much more than adequate. Tests we performed determined that the maximum utilization of the processor was only 17 percent while running the Learning Wall's laser pointer software.

The Learning Wall : Laser Pointer Software

The software to run the system is broken into three pieces. The first and most hardware-related software component, DETECT, performs low-level detection of the laser pointer in the camera's viewable area. There are two other applications that use the data provided by this program: INIT during the initialization process and LASERPEN during normal operation. Both of these helper applications run with a window title of LASERXY. All of the programs ask Windows to register a WM_LASER message through which communication takes place. All applications that request a message with that title will be returned the same unique identifier. Figures 2 and 3 illustrate the relationships between these programs.

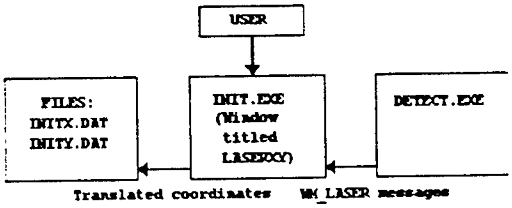


Figure 2. The Interaction between DETECT and INIT.

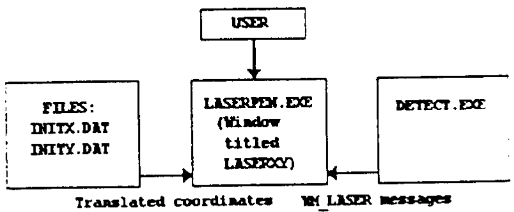


Figure 3. The Interaction between DETECT and LASERPEN.

The camera captures individual frames at 320 pixels in the x dimension and 240 pixels in the y dimension at as close to 30 frames per second as possible. Depending on how taxed the system is from user commands the frame rate can change. Each individual frame is processed to see if there was an instance of a pixel being brighter than a certain threshold. The environment that is present determines this threshold. DETECT can perform some functions as a stand-alone program, but it also cooperates with other applications through Windows messages. As a stand-alone and cooperative application it can display the output of the camera and also provide the tracking information of the laser pointer in camera coordinates. To improve speed, the tracking is performed by discarding half of the actual pixels that are visible to the camera. After an instance of a pixel over the threshold is found DETECT discerns if the pixel that was most recently skipped had levels of red, green and blue greater than the threshold. If so, the coordinates of the skipped pixel are used. This provides the accuracy of the entire resolution of the camera along with the speed of checking only half the pixels initially. Also, to improve efficiency, the program stops looking for an instance of the laser in a particular frame once it is seen. In order to run as a cooperative application, DETECT asks Windows to return the handle to LASERXY, the program that spawned it. This allows DETECT to send Windows messages to that specific application instead of a WM_BROADCAST message that could severely slow down the system. Once the camera distinguishes an instance of the laser pointer, it then sends messages to either LASERPEN or INIT with the laser pointer's position in the frame.

Sending the appropriate message involves knowing information about the previous frame as well. If in the previous frame an instance of the laser pointer wasn't seen, the program sends the coordinates along with a first_seen flag. If it was seen in the last frame, then it sends a continue_seen flag. When the laser pointer is not seen, but previously it was seen, the program sends (0,0) for the camera coordinates and a flag of not_seen. To minimize the number of Windows messages, the application doesn't continually send not_seen messages. Figure 4 is a state diagram showing the message states of DETECT. Figure 5 is a flow chart showing the logic involved in DETECT.

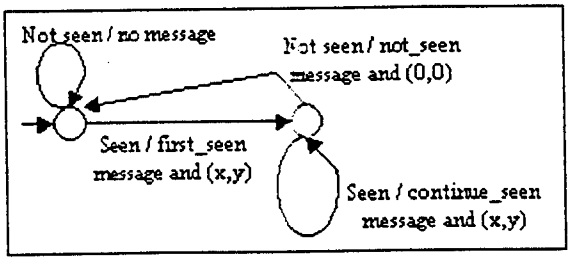


Figure 4. DETECT Message States.

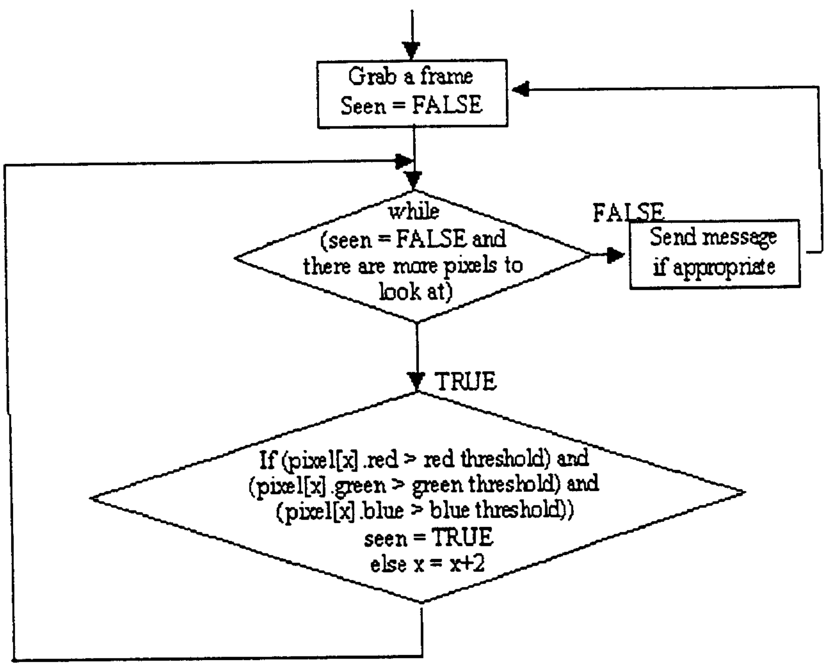


Figure 5. A High-Level Flow Chart of the DETECT Program.

__Menu Choices__

If DETECT, running as a stand-alone program, wasn't spawned by an application with the title of LASERXY, then it starts up in a normal window, displaying the captured frames along with the coordinates of the point where the laser pointer was seen. This can be toggled on and off by going to the *Image* menu and selecting *Toggle Output*. If an instance of the either INIT or LASERPEN is running, then the program toggles the output off and minimizes itself. The default camera position setting is front screen and can be toggled under the *Image* menu as well. While using the program for detection of the laser pointer, it must capture at 320 pixels by 240 pixels but for use as a stand-alone application, under *Image/Preview Size*, the user has many resolution options: 80x60, 160x120, 320x240, 480x360, 560x420 and 640x480.

Camera to Screen Transformation (INIT)

In a perfect environment the camera would be inside the projector and see exactly as much as the projector. In addition, the lens would be perfectly flat so that every pixel is the same size as every other pixel. Unfortunately this is not the case, so there is a need to tell LASERXY some specific information about the geometry of the camera and the image being projected. This is done through an initialization procedure in which two functions that produce a discrete display coordinate for every visible camera coordinate are generated:

x display coordinate = Fx (x,y camera coordinate)
y display coordinate = Fy (x,y camera coordinate)

The resulting transformation data is then saved onto disk for LASERPEN to read.

Upon running INIT the arrow icon is replaced with a series of four targets toward which the user should point the laser. These target points are in the four corners of the screen. Since the user's hand is not perfectly steady the software expects the user to point at each target and release the pointer when on target. The point where the laser was last seen is taken as the initialization point. The resulting four points are stored and used in the camera-to-screen transformation. What the camera sees of the projected image and the four target points (P1, P2, P3, P4) could be similar to what is shown in Figure 6. The projected image is 640 by 480 pixels, but in terms of camera viewable coordinates it is only 320 by 240 pixels.

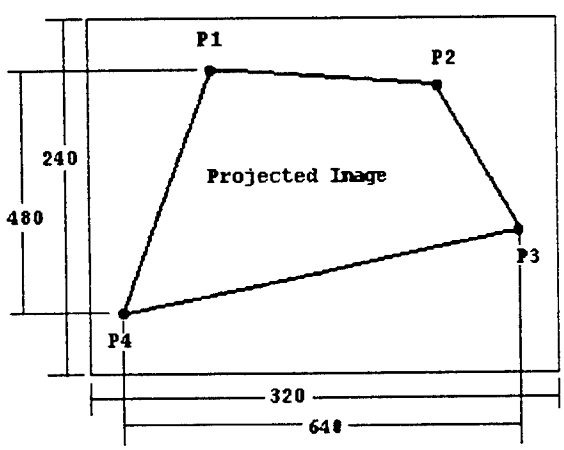


Figure 6. Simulated Camera Viewable Area.

First the intersection points of the "vertical" and "horizontal" lines bounding the area enclosed by the four target points are computed using the parametric equations for straight lines. Any point within the camera's viewable area is converted to its scaled screen coordinates by projecting a line from the point to each of the two previously calculated intersection points. As shown in Figure 7, those lines will each intersect with an opposing line. The intersection points are then stored and the result is evaluated as a percentage of the overall line and linearly scaled. In this example, the x values are scaled as a percentage of 640 pixels and the y values are scaled as a percentage of 480 pixels. This information is calculated for every point in the camera's viewable area and is stored in two two-dimensional arrays, one for x coordinate values and one for y coordinate values. At the end of the initialization program, the two arrays are stored onto the disk. This ensures that once a good initialization is calculated, it will never have to be repeated until there is a physical change in the setup.

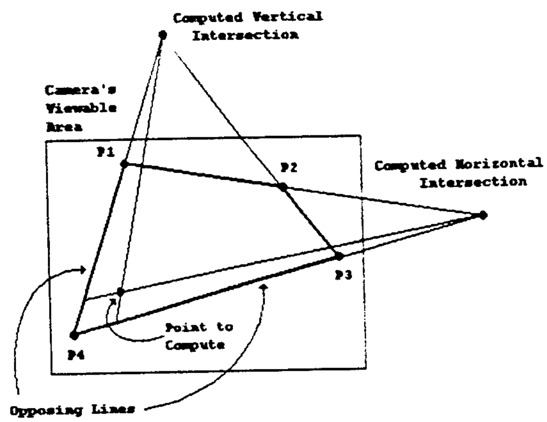


Figure 7. Camera to Screen Transformation.

The previous example dealt with displayed image resolutions of only 640 by 480 pixels. In practice the software can handle dynamic changes in resolution. Instead of calculating the points in reference to a display of 640 by 480 pixels, it calculates the points in reference to a display of 512,000 by 288,000 pixels. This is a common factor that can be evenly divided by 640 by 480 pixels, 800 by 600 pixels and 1024 by 768 pixels. By computing the points in this fashion the program doesn't have to know the resolution of the actual display, making it much more general and useful. This also allows calibration to occur regardless of whether the resolution of the screen is the same during calibration and normal operations. Figure 8 on the next page is a flow chart showing how y screen coordinates are computed from camera coordinates in 640 X 480 display mode.

LASERPEN

The final application that works directly with the laser pointer is called LASERPEN. This program is the meat and bones of the user interface since it listens to Windows messages and moves the pointer accordingly. The program begins by opening the files that were created by INIT and stores them into two arrays, one for the x-coordinates and one for the y-coordinates. After loading the files, a correction must be performed to change from the general x and y values to the x and y values for the current resolution. For resolutions of 640 by 480 pixels, the x and y values are divided by 800 and 600, respectively; for resolutions of 800 by 600, they are divided by 640 and 480; for resolutions of 1024 by 768, they are divided by 500 and 375. When a resolution change occurs, the program reloads the files from the disk and re-performs these calculations. By doing the calculations only once for each resolution change, the program removes two divisions that would otherwise need to be performed at every instance of the laser pointer inside the body of the loop.

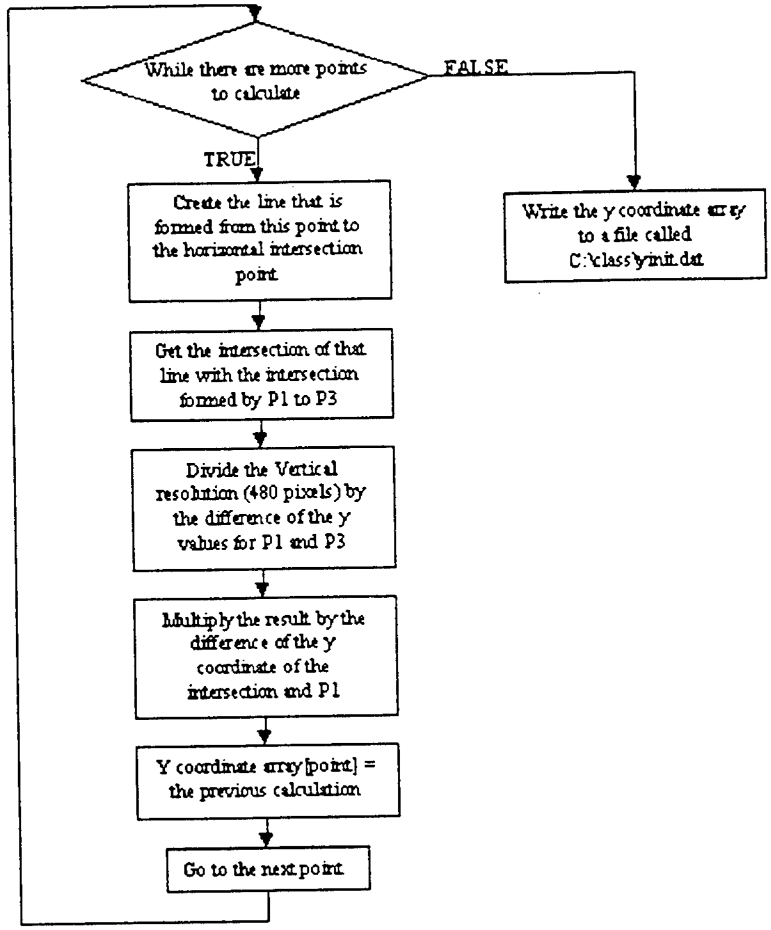


Figure 8. A High-Level Flow Chart Showing how the y Screen Coordinates are Computed For Each point in the Camera's Viewing Area.

LASERPEN doesn't open a standard window like most applications. Instead it creates a visible popup window in the lower right hand corner of the screen that is 25 pixels high and 50 pixels wide. This window has no borders, is solid white, and cannot be overwritten by other application windows. It is used to display the current emulation mode of the laser pointer (LEFT, RIGHT, or DRAW).

Windows and Visual C++ 5.0 provide functions that make generating button clicks fairly simple. The function mouse_event() takes as parameters the desired button to press and the location. In order to be able to generate double clicks, the mouse sensitivity under Windows needs to be lowered using the Windows Control Panel Mouse settings.

To alleviate the problem of user unsteadiness the following approach is taken: When the laser is turned on, an imaginary area equivalent to 1/16 of the overall horizontal resolution is drawn around the point. If the laser is detected again within one second of the last time it was detected and falls within this imaginary area, then LASERPEN generates the proper action. In LEFT mode a left mouse button mouse button down message is generated and, only when the pointer is not detected, is a left mouse button up message generated. In the RIGHT mode the same actions are performed but with the generation of right mouse button messages. If the program is in DRAW mode, instead of generating a button click, laser detection is used to generate the endpoint of a line segment. While the program still sees the laser pointer, the line is drawn to that point from its previous location. In all modes of operation the mouse pointer is instantly moved to its new location. This is much different from the case of a standard mouse in which the system has to take into account its current location before it can begin moving the mouse cursor. Figure 9 on the next page shows a high-level flow chart of the logic of the LASERPEN application.

All the required communication between these programs is done with WM_LASER messages. As seen in Figures 3 and 4, LASERPEN receives the WM_LASER message from DETECT. This message includes the relative location and the first_seen, continue_seen, and not_seen flags. When LASERPEN receives this message, a lookup is performed into the x and y coordinate arrays and the mouse is moved to that new location using the Windows function SetCursorPos().

The Wireless LAN Component

One way of enabling student laptop computers to communicate with the instructor's computer is to set up a wireless local area network (WLAN). Our WLAN consists of a Lucent Technologies WavePOINT II base station connected via a 10base2 cable to the server machine. The laptops then use type 2 slot PCMCIA WaveLAN cards with attached antennas that stick to the back side of the laptop. The base stations and the PCMCIA cards auto-connect when they are in proximity of each other and provide the laptops with access to the Intranet that the server provides. The LAN can provide up to 4 Mbps bandwidth broadcast at 2.4 GHz split among all users in the range of the base station. In an open area such as Binghamton University's lecture hall 12, the connectivity is limited to 200m, which is more than enough for even our largest lecture hall. The only requirement that the base station has is that it be mounted in plain view on the wall and that its antenna face the classroom.

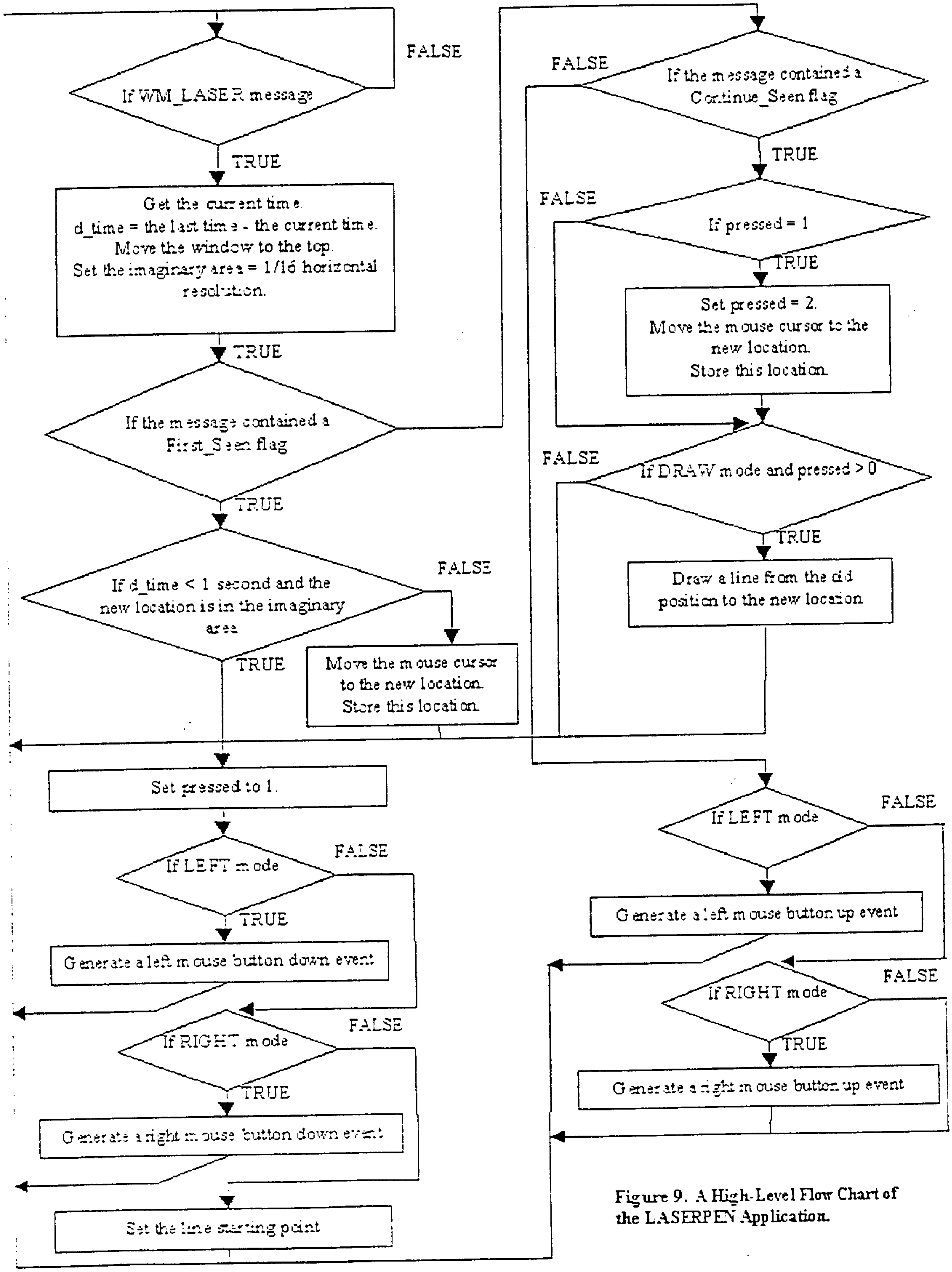


Figure 9. A High-Level Flow Chart of the LASERPEN Application.

There are no software modifications that have to be made to existing TCP/IP based applications as the Wireless LAN is merely an implementation that is transparent to the Operating System.

In order to allow students to control the instructor's mouse and keyboard over the WLAN, we created two custom pieces of Windows software: a server application (CLICKSERVE) and a client application (CLICKCLIENT) for each student laptop. These are Microsoft Visual C++, MFC, dialog-based applications. When a student starts his laptop computer, the wireless LAN will automatically connect the user to the intranet and provide him with an IP address for the TCP/IP protocol. This gives the student all of the facilities required to run nearly every web and Internet based application. If a student, connected to the network, runs our CLICKCLIENT application from his laptop, he can control the instructor computer's cursor and keyboard, thereby gaining control of the system console just as if it were his laptop. This type of interaction allows easy synchronization of students' questions to presentation material by allowing the students to easily point to the area in question. This also allows the students to change between slides, interact with software that is typically single user, and thus learn more efficiently.

Starting CLICKSERVE presents the instructor with a minimizable window that gives him the option of accepting or denying connections and allowing or denying student control over the system mouse and keyboard. This allows the instructor the flexibility of being able to decide when student intervention is most convenient or warranted.

When the student runs CLICKCLIENT, he is required to login to the server using his own unique username and password. This allows the server application to have more knowledge of the incoming requests and could be made customizable for different people. After pressing the "Start Capture" button, the student computer's screen is blanked white, and any mouse or keyboard actions he takes will be reflected on the instructor's computer screen. (The instructor also has the capability of controlling his own mouse cursor; mouse/keyboard messages from the instructor's system take precedence over messages coming over the local network from a student computer.) Pressing the <ESC> key on the student's laptop disables the student from controlling the system mouse and returns his screen to normal. CLICKCLIENT also allows the student, to "electronically" raise his or her hand. Pressing the <F1> "page" key on the laptop causes an audio file to be played on the server to notify the instructor that a student has a question. CLICKCLIENT and CLICKSERVE communicate through standard TCP/IP sockets. Whenever the user of the student computer performs any mouse or keyboard actions, CLICKCLIENT sends the appropriate Windows message and data parameters over the socket connection to CLICKSERVE.

Some of the mouse messages of interest are:

```
WM_*BUTTONDOWN      // A mouse button was pressed (* = L, R, or M)
WM_*BUTTONUP        // A mouse button was released (* = L, R, or M)
WM_*BUTTONDBLCLK    // A mouse button was double clicked (* = L, R or M)
WM_MOUSEMOVE        // User moved the mouse
```

The data that is sent over the network connection is identical to the information that is packaged in a normal Windows mouse event message--namely the x, y coordinates of the mouse cursor and codes indicating what action occurred and the status of the mouse buttons and certain keyboard keys (Shift, Ctrl, Alt) when the event occurred.

The keyboard events of interest are:

```
WM_KEYDOWN        // User pressed a keyboard key
WM_KEYUP          // User released a keyboard key
```

The data that is sent over the network for these kinds of events consists of the virtual key code that identifies the key pressed or released and other information related to the keyboard event that triggered the message.

Finally if the student presses his <F1> key (thereby attempting to "page" the instructor), the message is received as:

```
WM_USER+100       // User pressed his "page" key
```

The response of CLICKSERVE is to play a wave file on the instructor's computer.

The connection and communication between the student and instructor computers as well as the salient features of the CLICKCLIENT and CLICKSERVE programs are shown in Figure 10 on the next page.

By having the client send the information blindly and pass it on to Windows functions in the server that accept these parameters, the CLICKCLIENT and CLICKSERVE programs are general enough to sustain significant changes in the way Windows processes and packages mouse and keyboard events.

<u>The Voice Communications Component</u>

In order to add a further degree of freedom to the instructor, we attempted to add a voice command recognition component to the learning wall. The idea would be for the instructor to be able to control his application by voice commands over a wireless microphone as well as by using the laser pointer. We had hoped to be able to use an off-the-shelf speech recognition software package to control the kinds of applications an instructor would be using in the classroom (e.g., a Powerpoint presentation). We performed experiments with Dragon Systems Dragon Dictate for Windows and IBM's Via Voice, but, in both cases, ran into reliability problems. It seems that the kinds of voice recognition packages that are currently available are not capable of giving a sufficiently high percentage of successful commands recognized to be really useful in the kind of "noisy" classroom environment where we envision the learning wall would be used. A great deal of further research needs to be done in this area.

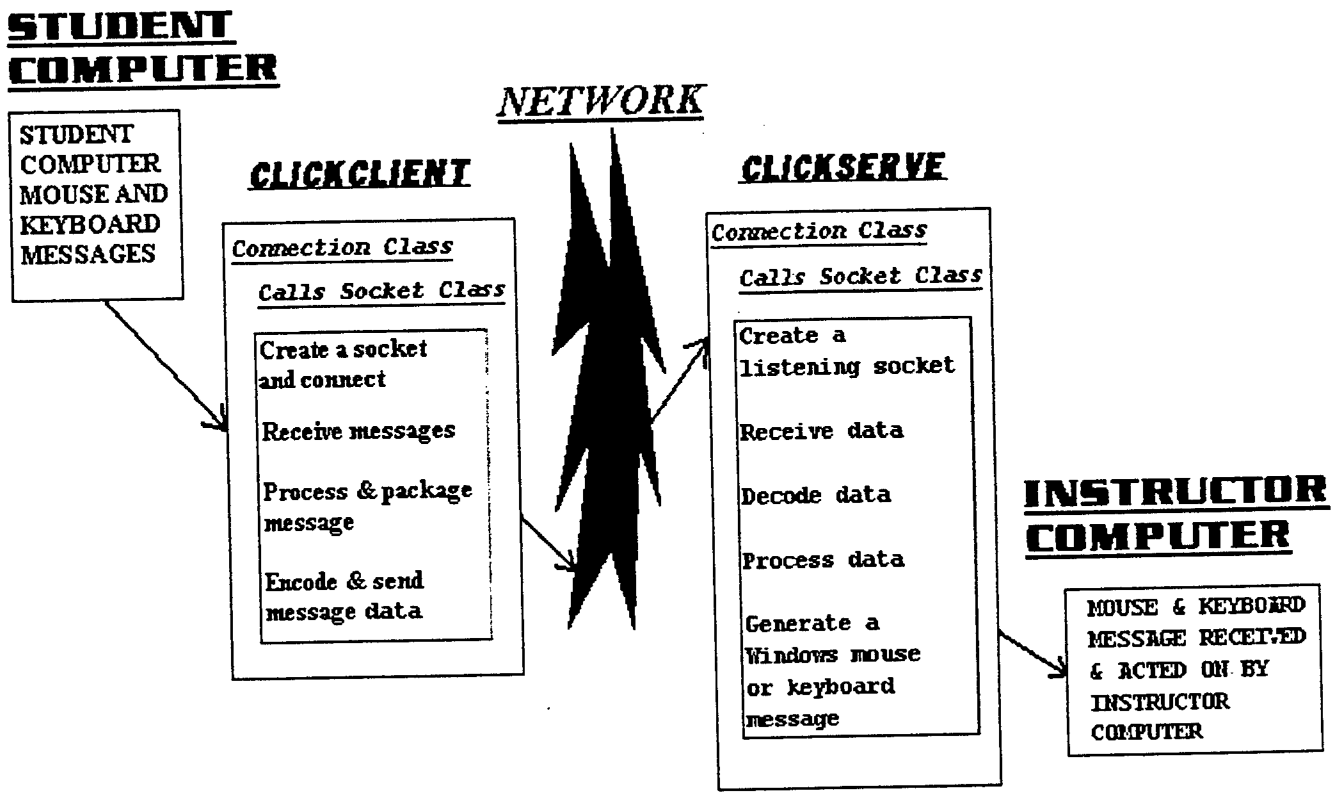


Figure 10. CLICKCLIENT and CLICKSERVE Controlling Windows Messages Over the WLAN Connection.

The Students

Because of the AFOSR grant, we were able to provide a real opportunity to do research in human-computer interfaces to two graduate students (Jason Moore and Veersen Ghatge) and six undergraduate students (Jaron Rubenstein, Richard Lao, Steaphan Greene, Brian Lalor, Thomas Happ, and Justin Terpstra). This group of students worked very well as a team and were able to successfully install, debug, and maintain all of the hardware we used in the project. They also designed, wrote, and debugged all of the related software. The graduate student who did the most of the work on the design and implementation of the laser pointer system, (Jason Moore) received his M.S. in Computer Science from Binghamton University in May, 1998, is now working for the Air Force Laboratory at Rome, New York on a project that is very closely related to our learning wall project. Eventually he is planning to continue studies toward the Ph.D. degree at Binghamton University in the area of human-computer interaction. One of the undergraduate students involved in the project (Steaphan Greene) worked last summer at Rome Lab. He and one of the undergraduates who joined our group this semester (Justin Terpstra) are both considering working at Rome Lab this coming summer.

Possible Publications

We are in the process of writing papers describing our results. Our plan is to try to submit them to one of the ACM's SIGCHI (Special Interest Group on Computer Human Interaction) publications. One of the papers will be on the laser pointer system and the other on the CLICKCLIENT/CLICKSERVE wireless LAN system.

Conclusions

At Binghamton University several people have tried using our Interactive Learning Wall system. In all cases the learning curve required to become productive was very small. The time needed to become proficient was spent mostly in becoming familiar with how the laser pointer's "click" sequence works. After becoming comfortable with that, the interface proved to be quite intuitive. On the basis of our initial tests, we feel that this type of device could be used effectively at other institutions.

References

[1] www.rl.af.mil/programs/ADII/adii_dw.html

[2] Moore, J.A., "Interactive Learning Wall," Binghamton University Masters Thesis, 1998.

INSTITUTE FOR SIMULATION & TRAINING

AFOSR Summer Research Extension Program

Web-Based Distributed Simulation

Sponsor: Air Force Office of Scientific Research

Contract Number: 98-0822

P.I.: Dr. Kuo-Chi Lin

Prepared by:
Institute for
Simulation & Training
University of Central Florida
Orlando, Florida

Prepared for:
Air Force Office of
Scientific Research

University of Central Florida

IST-TR-98-14

WEB-BASED DISTRIBUTED SIMULATION

Kuo-Chi Lin
Institute for Simulation and Training
University of Central Florida
Orlando, FL 32826

Abstract

The multi-user virtual environment is a computer-generated synthetic world on the network where users can navigate and interact with each other in the same virtual world. The World Wide Web (WWW) environment is a client/server architecture composed of various server sites to form a global database web over the Internet. Since the existing WWW browsers have difficulty to create an interactive multi-user virtual world, the SharedWeb system, a three-dimensional web browsing system, is designed to support the interactions among the clients over the WWW environment. This report describes the design and implementation of SharedWeb.

The SharedWeb system has the following features that are necessary for the seamless integration between a multi-user interactive system and the WWW environment. 1. The user can download a scene file from any WWW server with the HyperText Transfer Protocol (HTTP). 2. The SharedWeb system provides the hyperlink feature to retrieve various media resources that are supported by the WWW environment. 3. The SharedWeb system can handle information exchange between HTML documents and the objects inside a virtual world. 4. The SharedWeb system server itself is an add-on of an existing WWW server. 5. The user can directly enter a virtual world through the WWW server from which the scene file is downloaded.

The SharedWeb system has been successfully tested in both the Local Area Network (LAN) and Wide Area Network (WAN) environment. The SharedWeb system uses the Dead Reckoning technique to reduce the network traffic created by the interactions among the users, and to maintain the interactivity even when the

congested Internet delays the PDUs. After receiving a delayed PDU, proper delay compensation and smoothing are needed to make the interaction realistic.

1. Introduction

Virtual environment is a technique to immerse the user into a synthetic world built by the computer or a remote environment electronically transmitted to the computer for the purpose of training, simulation, prototyping, design, or data acquisition. Due to the rapid development of the virtual environment technique, the multiple-user virtual environment has become a popular topic over the years. The multi-user virtual environment is a computer-generated synthetic world on the network where users can navigate and interact with each other in the same virtual world. The gaming industry has developed many multi-user systems over the Internet. An extensive survey of these existing systems can be found in Section 2.

The World Wide Web (WWW) is a project that brings a global information universe to the existence. [1] The Virtual Reality Modeling Language (VRML) project is an attempt to deliver a virtual world into the WWW environment. [2] The original goal of designing VRML language is to allow multiple participants to interact over the Internet. After the introduction of the VRML language, many VRML-supported browsers were developed, which include the WebSpace from SGI, WebOOGL from the Geometry Center, AmberGL from DIVE Lab., WorldView for Intervista Software, Inc. With these browsers, you can view and walk through a virtual world described by the retrieved VRML file. However, none of these browsers has fulfilled the original goal of multiple-user interactivity, even for those browsers that have implemented VRML version 2.0 released in August 1996.

In this report, the author will describe the design and implementation of

SharedWeb, a web browsing system that can support the interaction among clients over the existent WWW environment.

2. Multi-user Browser System

Utilizing the existing WWW environment to build a multi-user interactive system, new communication protocols need to be added to provide the interactions among distributed web browsers. Instead of modifying the existing HTTP protocols WWW is using, the goal is to design a new multi-user virtual system and seamlessly integrate it with the WWW architecture. For a distributed multi-user system to be seamlessly integrated with the WWW environment, it must have the following characteristics: [3]

1. The user can download a scene file from any WWW server with the HyperText Transfer Protocol (HTTP).
2. The multi-user system must provide the hyperlink feature to retrieve various media resources that are supported by the WWW environment.
3. The multi-user system must be able to handle information exchange between HTML documents and the objects inside a virtual world.
4. The multi-user system server itself is an add-on of an existing WWW server.
5. The user can directly enter a virtual world through the WWW server from which the scene file is downloaded.

In summary, the seamless integration implies that the user can simply download a virtual scene file from the supporting WWW server, navigate through the virtual worlds managed by the WWW servers, and fully utilize the services provided by the

WWW environment without the user's awareness.

The researches on the multi-user virtual environment can be classified into two categories: network-based and web-based virtual reality systems. The network-based system researchers generally focus on providing a large-scale environment with every available network techniques. On the other hand, the web-base system researchers emphasize on providing multi-user virtual environment on top of the existent WWW environment. Although our work is concentrated on designing a multi-user virtual reality system over the existent WWW environment, the techniques from the network-base multi-user system also provide partial solutions to our work.

The WWW environment is a client/server architecture using HTTP protocol. Due to the intrinsic properties of HTTP, the WWW system is a one way street that can not achieve interactivity among users. Hence, new mechanisms and protocols have to be designed to support the multi-user interaction over the WWW environment. The VRML standard is an attempt to deliver the virtual world on the WWW environment, and its original goal is to support the interaction of multiple participants. However, even the VRML version 2.0 [2] failed to define the standard of the multiple user interactions. Therefore, several designers of the web-based multi-user systems have developed their own proprietary methods to support the multi-user interaction. [4~5]

Virtual Society (VS) is a research project of Sony Computer Science Laboratory, Inc. in attempt to define a global architecture and a set of protocols to realize a multi-user interactive 3D environment in a WWW setting. [4] It was originally based on the DIVE platform developed by the Swedish Institute of Computer Science. [6] The VS project has developed a VRML-standard virtual reality system over the WWW environment, called Community Place Browser. [7] The VS architecture, however, is not seamlessly integrated with the WWW environment. For example, the VS client can not enter a virtual world from the WWW server from which the scene file was downloaded. After a scene file is downloaded from a WWW server, the VS client must send a URL to that WWW server to query the necessary information, such as IP address and port number, of the VS server. In addition, the VS server does not support information exchange between the HTML document and objects inside a virtual world. Hence, VS is an extension of the networked virtual reality system to the WWW environment rather than a merge of the two.

The German National Research Center for Information Technology (GMD) has proposed another type of web-based multi-user system by defining new nodes for VRML standard. [5] Different from the VS project, the multi-user system by GMD does not require a multi-user server to maintain the consistency among distributed objects. To achieve the consistent control of shared objects, GMD proposed a so-called Active Lock mechanism and two extension nodes, Interaction node and Behavior node, for the VRML standard. The research of GMD has been concentrating on extending the VRML standard for the multi-user interaction. Issues of seamless integration discussed in the previous section still require further study.

Scalable Platform for Large Interactive Networked Environment (Spline) by Mitsubishi Electric Research Lab. (MERL) uses yet another approach to integrate the multi-user virtual environment with the WWW environment. [8] Spline is essentially a networked-based multi-user virtual reality system that uses Interactive Sharing Transfer Protocol (ISTP) as its communication protocol. [9] ISTP incorporated several simple HTTP protocols to enable the HTTP server to participate in multi-user interactions. With the help of ISTP, the Spline system provides a WWW-like virtual environment. However, it is not a web-based multi-user system.

Other web-based multi-user systems that are commercially available include Ccpro [10] from Blaxxun interactive Corp., OnLive! Community browser [11] from OnLive! Technologies Inc., OZ Virtual [12] from OZ Interactive Inc. and V*Realm Multi-user browser [13] from Integrated Data Systems, Inc. All of the above browsers adopt the VRML standard as their virtual scene file format. Each of the above multi-user systems uses its own technique to provide interactivity among distributed users. None of the above browsers is fully integrated with the traditional HTML browser. Active World browser [14] from Worlds Inc. is another kind of multi-user browser that is not a VRML browser, whereas it merges the traditional HTML browser into its own browser. However, it can not handle information exchange between HTML documents and the objects inside a virtual world.

3. Design Considerations

Since the existing WWW browsers have difficulties in creating an interactive multi-user virtual world, the SharedWeb system, a three-dimensional web browsing system, is designed to support the interactions among the clients over the WWW environment. [15]

3.1 Mechanisms to Support the Multi-user Interactivity over WWW

There are four intrinsic problems must be solved to support the multi-user interaction over the WWW environment. They are Client-Information Recording, Server-to-Client Callback, Excessive Network Load, and Virtual World Entrance.

3.1.1 Client-Information Recording

The WWW server uses a request-and-response technique to serve its client's request. The link between the server and client is established only when a client issues a request to the server. After the requested information is sent, the link is broken and the client is "forgotten" by the server immediately. However, in a multi-user interaction system, the server needs to keep track of all the users. Since the link is broken immediately after the requested information is sent to the client, the WWW server does not keep the records of the clients who have requested information. We call this the Client Information Recording problem.

To solve this problem, each browser must be able to provide a sequence number, called Unique Identification (UID), to uniquely identify itself to the server. Modified from Chin's proposal [16~17], the SharedWeb browser uses the combination of Login name, Time stamp and listen Port number (LTP) as the UID to identify itself. The login name is the nickname of the user to login to a virtual world, the value of the time stamp is the time when the browser was first activated and the listen port number is the port number that the browser used to receive messages. With this UID number, the SharedWeb server can easily identify a new registered user and assigns a User ID to that browser. Moreover, the SharedWeb server will record this UID number for future reference.

3.1.2 Server-to-Client Callback

The link between the server and the client is established only when a client issues a specific request. The WWW server does not actively set up a connection to its clients; therefore, it does not have the ability to forward information from one client to other clients. This is referred as the Server-to-Client Callback problem.

To solve this problem, the SharedWeb server is designed to have the ability of actively set up connection the clients and forwarding information from one user to other users.

Another technique being widely discussed recently is "Cookies". [18~19] The Cookie technique is based on a two-stage process. First, a "cookie", or a small piece of information, is sent by the Web server and stored in the user's computer without its consent or knowledge. Second, the cookie clandestinely and automatically establishes connection to the Web server. In its current form, Cookies can not be used as the Callback mechanism from WWW server.

3.1.3 Excessive Network Load

Since the link between WWW server and client is established only when a request is issued, in addition to the intrinsic network latency, there is a delay associated with establishing the link. A central server is usually used in a web-based multi-user virtual system to mediate the communication among the clients. [5] The network bandwidth will be overloaded very quickly as the number of users increases, and the server will rapidly become an input/output bottleneck. We describe this situation as an Excessive Network Load problem.

Distributed Interactive Simulation (DIS) [20] is a technique developed by the Department of Defense for military training. It is a fully distributed simulation environment with no centralized control. In DIS, a concept called Dead Reckoning is employed to reduce the network traffic problem. [21~22] The Dead Reckoning model uses the current position and velocity acceleration to predict the future positions. All entities use the same dead reckoning model to update the position of one particular entity to achieve consistency. Each entity is responsible for comparing its own dead reckoning model with the true model. When the error exceeds a predefined threshold value, it needs to inform all entities its new position. SharedWeb can adopt the same concept. The states of all avatars are updated based on the respective dead reckoning models until the new status change information of one particular avatar is received. The information is sent in the form of a self-contained data package, called Protocol Data Unit (PDU), and forwarded to each participating browser through the SharedWeb server. The PDU is sent to the SharedWeb server only when the deviation of the dead reckoning model from the true model exceeds a predefined threshold. Hence, the bandwidth requirement can be reduced.

3.1.4 Virtual World Entrance

For a communication protocol to support the multi-user interaction over the WWW environment, the seamless integration is important for an adopted virtual environment to take full advantage of the WWW environment. The user has to be able to set up connection to the server without hassle. The problem is referred as Virtual World Entrance.

To solve this problem, we can either embed the multi-user server information into the downloaded scene file or allow the client to issue a query message to the WWW server for the information of the multi-user server. Although the first option is simpler, this method will cause trouble when a multi-user server is crashed and a replacement server is activated. As to the second option, the client has to wait until the replied HTML document is received before it can be connected to the virtual world as Virtual Society from Sony SCL did. [23] Both methods fail to satisfy the criteria of the seamless integration as the outline in Section 2.

The WWW server has to treat the multi-user server as an add-on application in order to integrate the network virtual system into the WWW environment seamlessly. In the WWW architecture, the WWW server uses the Common Gateway Interface

(CGI) to communicate to its add-on application. [24] Hence, a CGI program can be implemented to forward client information from the WWW server to the multi-user server. Using this approach, whether the multi-user server and the WWW server reside in the same host does not matter to the client. In addition, a hierarchy of the multi-user servers can be accordingly implemented. The WWW server passes the login information of a client to a load-monitor program, which will in turn forward the client to one of the multi-user servers without the awareness of the client. In this case, the multi-user server system can have a better load balance and fault tolerance.

3.2 Time Synchronization

Time synchronization is another important issue in the multi-user virtual system. The multi-user virtual environment is a synthetic world on the network where users can navigate and interact with each other. In our real world, all people use the same standard time to synchronize their local clocks and interact with each other. In the virtual world, however, the users join the scene from different computers, and the time in each user's virtual world is decided by the respective computer's system time. The time needs to be concordant for all users in the same virtual world so that the event actions are referring to the same clock. It is the Time Synchronization problem.

The Network Time Protocol (NTP) has been used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. It provides client accuracy typically within a millisecond on LANs and up to a few tens of milliseconds on WANs relative to a primary server synchronized to Coordinated Universal Time (UTC) via a Global Positioning Service (GPS) receiver, for example. Typical NTP configurations utilize multiple redundant servers and diverse network paths, in order to achieve high accuracy and reliability.

Some configurations include cryptographic authentication to prevent accidental or malicious protocol attacks. [25]

The NTP architecture, protocol and algorithms have evolved over almost two decades, with the latest NTP Version 3 and 4 designated an Internet standard. Among the goals of this design are: [26]

1. Optimize the computer clock accuracy and stability, subject to constraints of network overheads and/or telephone toll charges, relative to local and/or remote sources of time.
2. Enhance the reliability by detecting and discarding misbehaving local and/or remote sources and reconfiguring network paths as necessary.
3. Automatically adjust algorithm parameters in response to prevailing network delay/jitter conditions and the measured stability of the computer clock oscillator.

The NTP technique can help the SharedWeb system to achieve time synchronization. All users synchronize their time with SharedWeb server using the NTP technique before they enter the global virtual world. It can ensure that one event happens at the same time for all users in the same virtual world.

4. The SharedWeb System

With the mechanisms presented in the previous section, the SharedWeb system provides a natural extension of the WWW environment to a multi-user virtual environment. The infrastructure of the SharedWeb system is shown in Figure 1. The nodes of the SharedWeb system can be classified into two types: the server site and the browser site. The browser site is composed of four modules: Multiple Participants Interface, 3D Render Engine, Chat Phase, and WWW Homepage Viewer modules. The server site is built on top of the existent WWW server with two modules: a CGI program and the SharedWeb server. The server and the browser are communicated by the SharedWeb Communication Protocol (SWCP), which will be fully explored in the next subsection. Each of these modules is responsible for the tasks described as below:

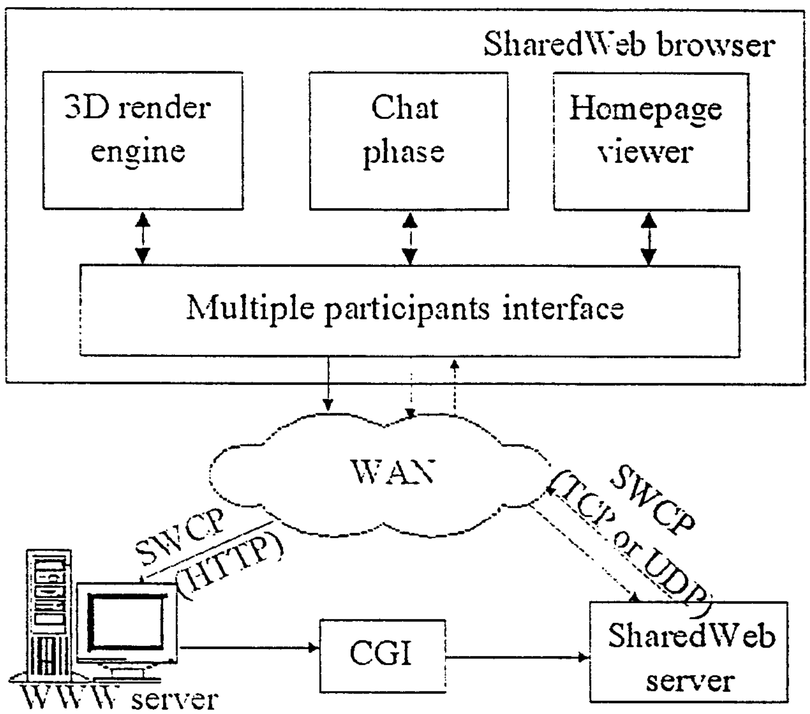


Figure 1. Infrastructure of the SharedWeb system.

4.1 The Browser Site

The Multiple Participant Interface module runs on top of Winsock to manipulate the interaction between the browser and other users. This module is the kernel of a SharedWeb browser. The main functions of this module are to calculate browser's

UID, encapsulate and de-capsulate the PDUs, transmit/receive data to/from the network, and coordinate messages among 3D Render Engine, Chat Phase, and WWW Homepage Viewer modules.

The 3D Render Engine module provides a visual window for the user to explore the virtual world. When a browser retrieves a scene file from the WWW server, it is an ASCII file. The scene file will then be translated into a virtual world database recognized by the 3D Render Engine module.

The Chat Phase module is designed to support text communication among distributed users. This module enhances the interactions among distributed participants and, hence, broadens the applicability of the SharedWeb system. Without this module, the shared virtual world will be a wordless world that makes the user bored easily when playing with the SharedWeb browser.

The WWW Homepage Viewer module is a standard HTML browser. By seamlessly integrating this module into the SharedWeb browser, a user can enter another virtual world by double clicking a hyperlink provided on a Homepage or inside the virtual world. Further, the user can exchange information with each other by using the Homepage and the HTTP protocol.

4.2 The Server Site

The WWW server is a gateway for a SharedWeb user to enter a virtual world. For the user, he only needs to download a scene file from this WWW server and to login his nickname to enter a virtual world. After the WWW server receives the login information from a browser, the WWW server will pass the received user information to the SharedWeb server through the CGI program mentioned in the previous section.

The SharedWeb server module is the kernel of the entire SharedWeb system.

This program is responsible for mediating the information exchange among the browsers in the same virtual world. That is, all the client information is sent to the server and the server is responsible for routing the received information to the browsers in the same virtual world, with the exception of the chat (see Chat PDU below) among users.

4.3 SharedWeb Communication Protocol (SWCP)

There is no standardized communication protocol for the web-based multi-user virtual system. Different protocols have been proposed by various systems over the years. [4~5][9] The communication protocol includes steps to join a virtual world, to forward messages among distributed users, and to leave the virtual world. On the other hand, the DIS standard is by far the most successful technique that enables a fully distributed virtual environment to be built on a network. The DIS standard [20] defines the data format, PDU, and the communication protocol that employs various PDUs to achieve the spatial consistency among distributed users.

Although the DIS standard has been proven to be an effective method to support the networked multi-user virtual environment on the Internet, it is not designed for the web-based multi-user virtual environment. Inspired by the DIS technique, the SharedWeb system adopts the concept of PDUs to provide the multi-user interaction over the WWW environment. The PDU design focuses fully on the integration of the virtual system into the WWW architecture. Seven PDUs are defined in the SharedWeb system: [27]

1. Initial PDU: This PDU is used to submit the information required to log into a virtual world, and to inform other users that a new user just joined.

2. Acknowledge PDU: When the SharedWeb server receives an Initial PDU from a browser, it will respond with an Acknowledge PDU to the browser.
3. End PDU: When a user wants to exit from the virtual world, it will send an End PDU to notify the SharedWeb server and all other users.
4. Object State PDU: Browser uses this PDU to send the status-change information to the SharedWeb server and other browsers. To reduce network traffic, this PDU is issued only when the dead-reckoned position of the avatar deviates from its actual position by a predefined threshold value.
5. Chat PDU: SharedWeb system supports the peer-to-peer chatting protocol. This is the only PDU that bypasses the SharedWeb server. It can prevent the bottleneck on the SharedWeb server site as the number of user increases.
6. HTTP PDU: This PDU has two functions in this system. First, this PDU allows the user to send a URL address of a network resource, such as an HTML document or a audio clip, to other users. Second, it lets the users control the objects in the virtual world from an HTML document. Whenever data are filled into an HTML document provided by the SharedWeb browser, this information is sent to its own Web server, as shown in Figure 1. The CGI decodes the document and forward the information to the SharedWeb server. The SharedWeb server then packs the information into an HTTP PDU and forwards to other browsers.
7. Error PDU: When an error occurs between the clients and server, users involved are informed using this PDU.

4.4 User Interaction with the SharedWeb System

The user interactions in the SharedWeb system include six steps: Initialization, Login, Update, Chat, HTTP, and End. [28] The SharedWeb browser's start-up page is shown in Figure 2.

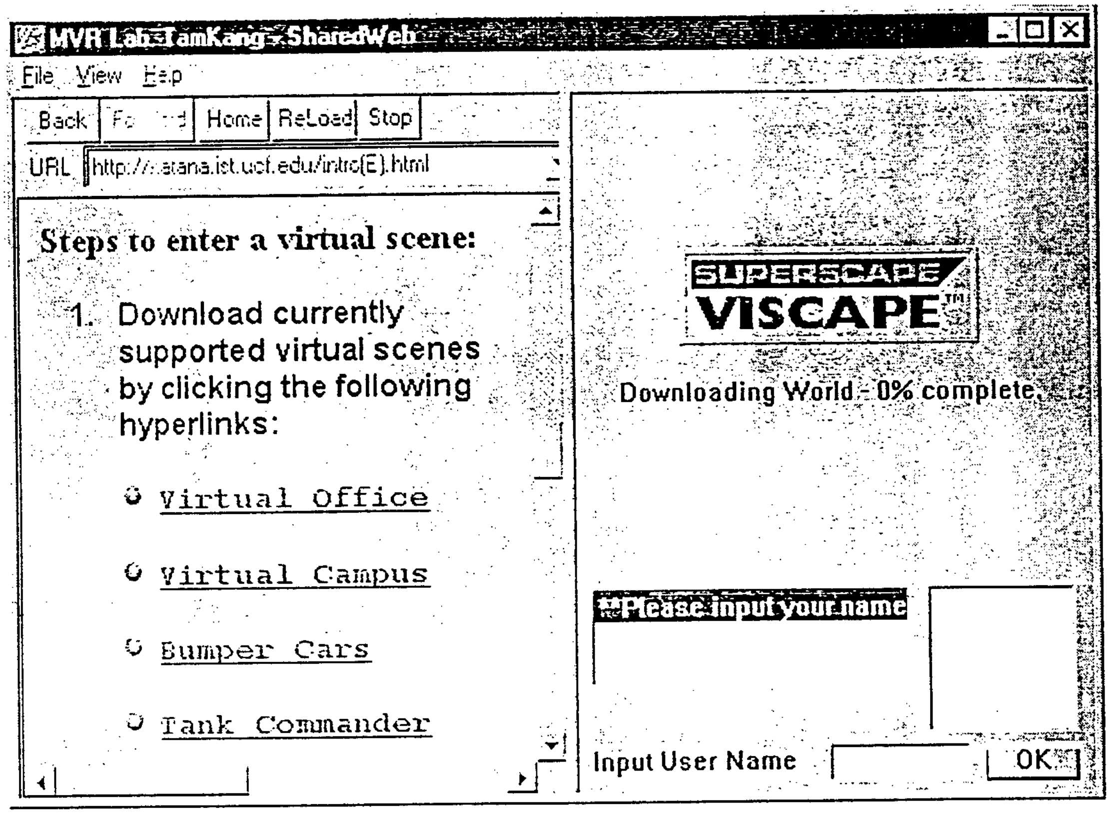


Figure 2. Start-up page of the SharedWeb browser.

4.4.1 Initialization step

This is the first step a user joins the multi-user system. Once launched, the SharedWeb browser (client) shows a homepage with a list of exercises available, as shown in the left window in Figure 2. The user selects the exercise and downloads the virtual world scene file and the related information document.

4.4.2 Login step

The user keys in a nickname of his/her choice in the lower-right "Input User Name" window in Figure 2. After the SharedWeb server receives the Initial PDU, it will register the user and send an Acknowledge PDU back to the SharedWeb browser.

Figure 3 is an example of the browser after the registration is completed.

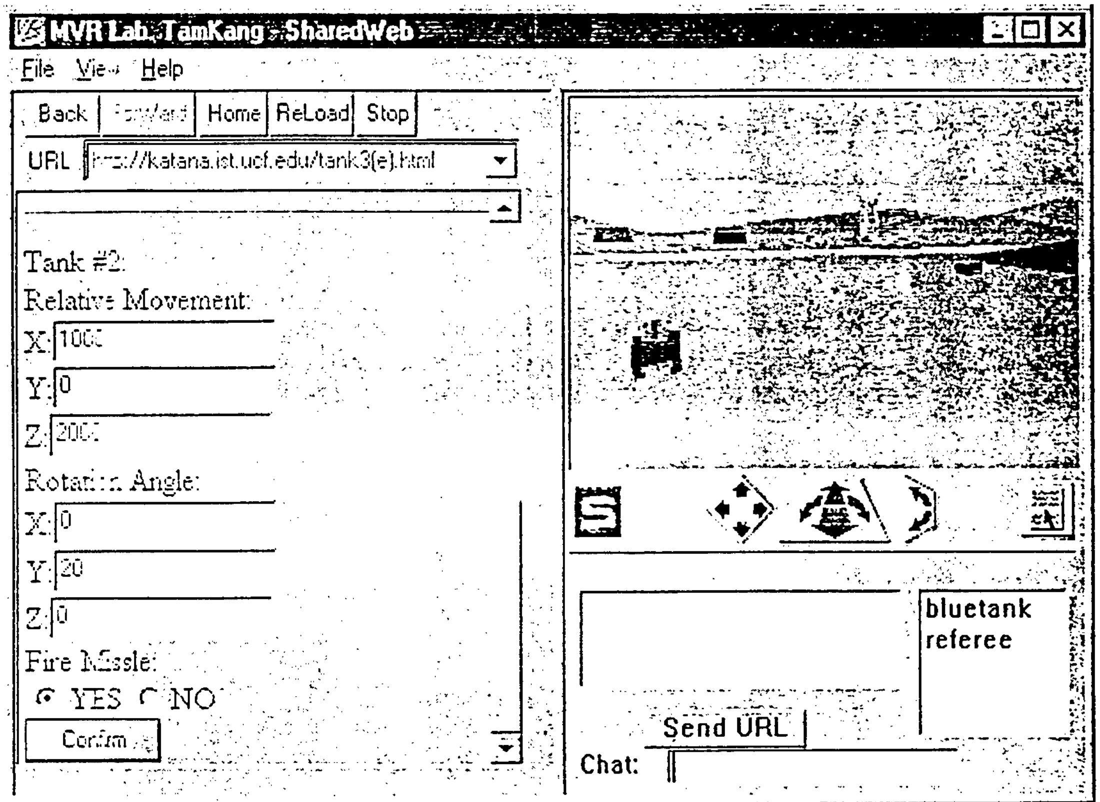


Figure 3. Viewpoint of the red tank commander after registration.

4.4.3 Update step

The user now can start to navigate the avatar in the virtual world. The SharedWeb browser provides a control bar and an HTML form, as shown in Figure 3. The user can use either one to control the avatar. When the dead-reckoned position of the avatar deviates from its actual position by a predefined threshold value, the browser automatically sends an Object State PDU to the server to inform other users in the same virtual world its update position.

4.4.4 Chat step

The user uses a chat window, as shown in Figure 3, to send text messages to any or all users in the same virtual world. The SharedWeb has adopted the Internet Relay Protocol [29] in its Chat PDU design.

4.4.5 HTTP step

The SharedWeb system can integrate Web resources into the virtual world. This is achieved by implementing the HTTP PDU. The user can send a URL address to other users in the same virtual world. The SharedWeb browser receiving the HTTP PDU will auto-download this HTML document from the addressed WWW server.

4.4.6 End step

The user can exit from the virtual world by select "Exit World" under the "File" menu. The browser sends an End PDU to the SharedWeb server.

5. SharedWeb System Implementation

The SharedWeb system is implemented using Microsoft Visual C++ version 5.0. The SharedWeb server runs on Windows NT Server version 4.0 with an embedded WWW server. The SharedWeb browser is currently available as a standalone browser. In the future, it will be implemented as a plug-in application for the Netscape Communicator and Internet Explorer. The user interface of the SharedWeb browser is shown in Figure 2 and 3.

As displayed on the upper-right window of Figure 2, Viscape from Superscape Inc. [30] is used as the 3D Render Engine. When the browser is initiated, the user will get SharedWeb system information in the left HTML window. After the user has successfully logged in a virtual world, the Chat Input Box will be activated automatically and all of the users' names inside the virtual world will be listed in the

User List Window. Further more, a "Send URL" button will emerge at the same time, as shown in Figure 4. The "Send URL" button is a special function for the user to send a hyperlink resource, such as an HTML document or audio clip, to other users. This function is another feature for the SharedWeb system to be seamlessly integrated into the WWW environment.

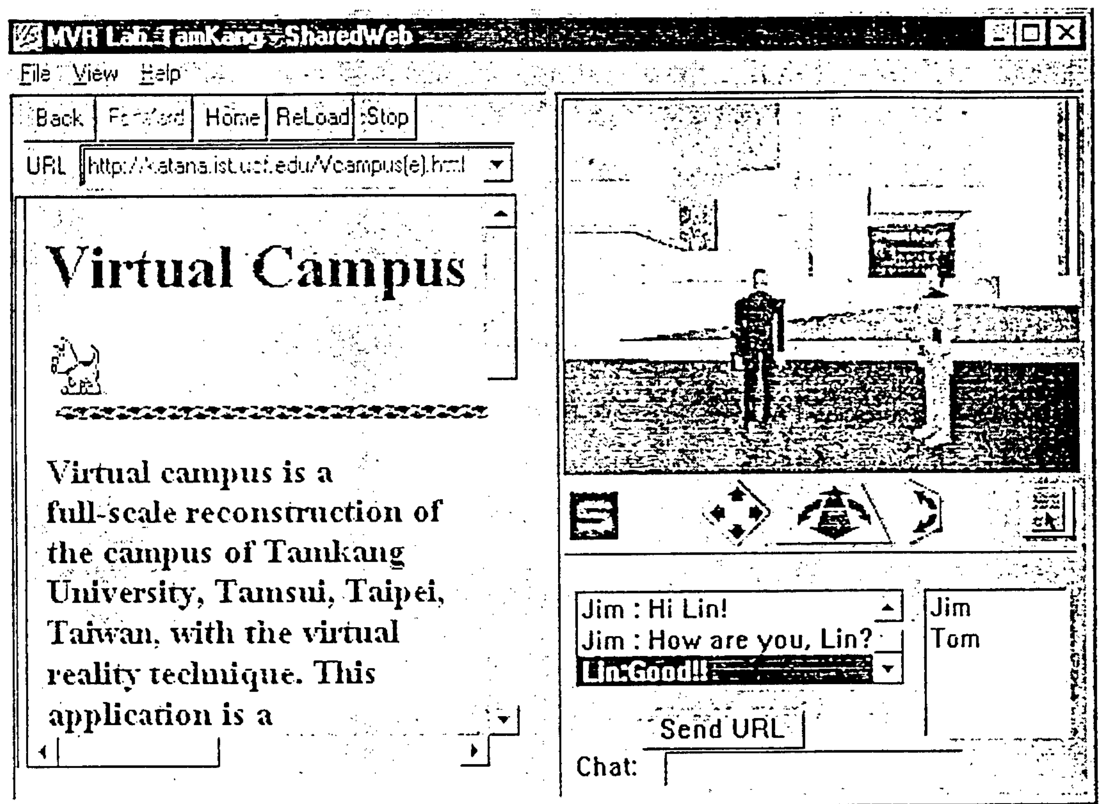


Figure 4. A scene in the virtual campus.

After the "Send URL" button is clicked, it will disappear and a highlighted message will be displayed in Chat Output Window to notify the user. In the meantime, the label of the Chat Input Box is changed to "URL:" for the user to input the URL address of a network resource, such as an HTML document or audio clip. Moreover, the sender selects the receiver(s) by clicking one or more name(s) from the

User List Window. By clicking the OK button, the specified network resource will then be automatically forwarded to the receiver(s).

To fully explore the features provided by the SharedWeb system, four virtual worlds have been constructed. The following scenarios are the virtual worlds currently supported in the SharedWeb environment. The user can move from one scenario to another without logging out.

5.1 Virtual Office

Virtual Office is an application of the SharedWeb system to demonstrate the benefits of integrating an HTML browser with a 3D browser. The scene of this virtual world is shown in Figure 5. The scenario of this demonstration is the Office of Administration. The user can request an application form in the HTML format. The form requested will appear on the left window in Figure 5. After filling the form, the user can submit it by clicking the "Confirm" button. The chatting capability in this demonstration provides a means for the student to ask questions in case he/she has any questions about the procedures.

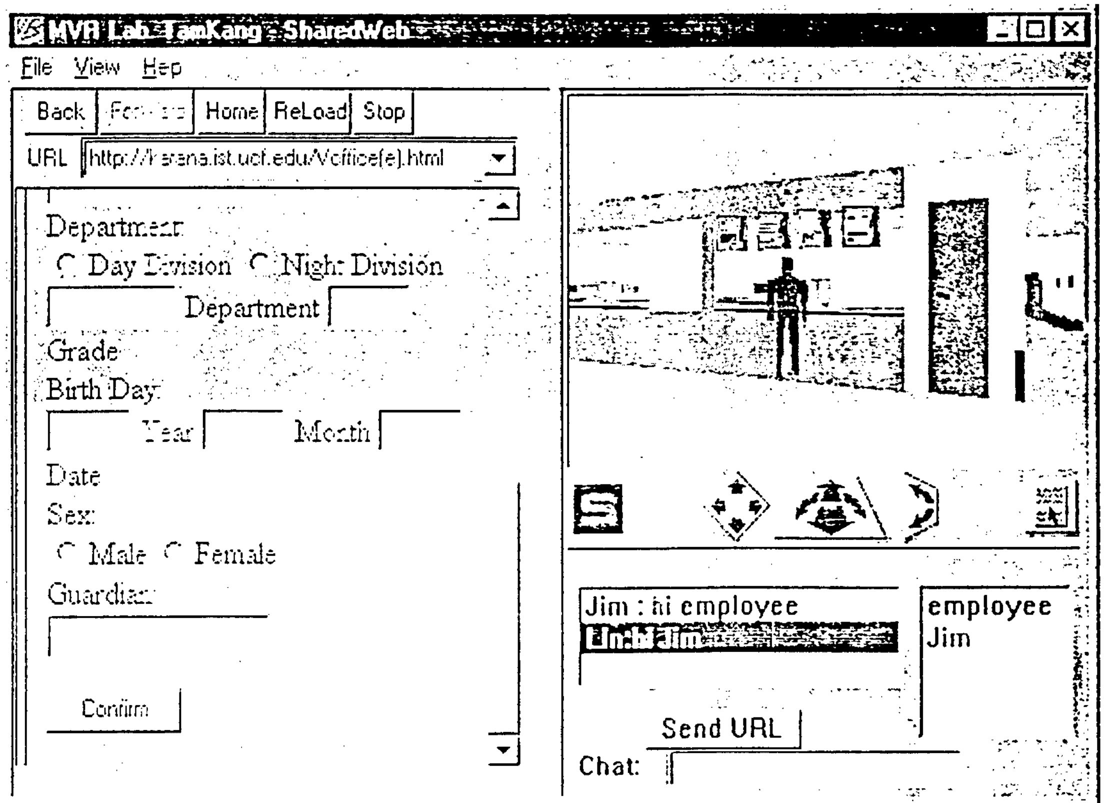


Figure 5. A scene in the Virtual Office and the form requested in HTML form.

Three methods are provided by the SharedWeb system for students to access the application form. The first method is to ask a virtual employee, who also logs in via a SharedWeb Browser, in the virtual office to send him/her the application form. The second way to retrieve the application form is to click a virtual photo on the wall that represents a particular application form. The third way for the student to access any application form is to click the HTML hyperlinks.

5.2 Virtual Campus

The virtual campus is modeled with the campus of Tamkang University in Taiwan. The user can walk through the campus and access to different virtual buildings and virtual offices. Figure 4 in Section 5 shows an example of the scenes the user sees in the virtual campus. The Office of Administration in scenario 5.1 is one of the virtual offices the user can enter.

5.3 Bumper Cars

This is a game of bumper cars. Each user controls his/her own car and bumps with other cars, as shown in Figure 6.

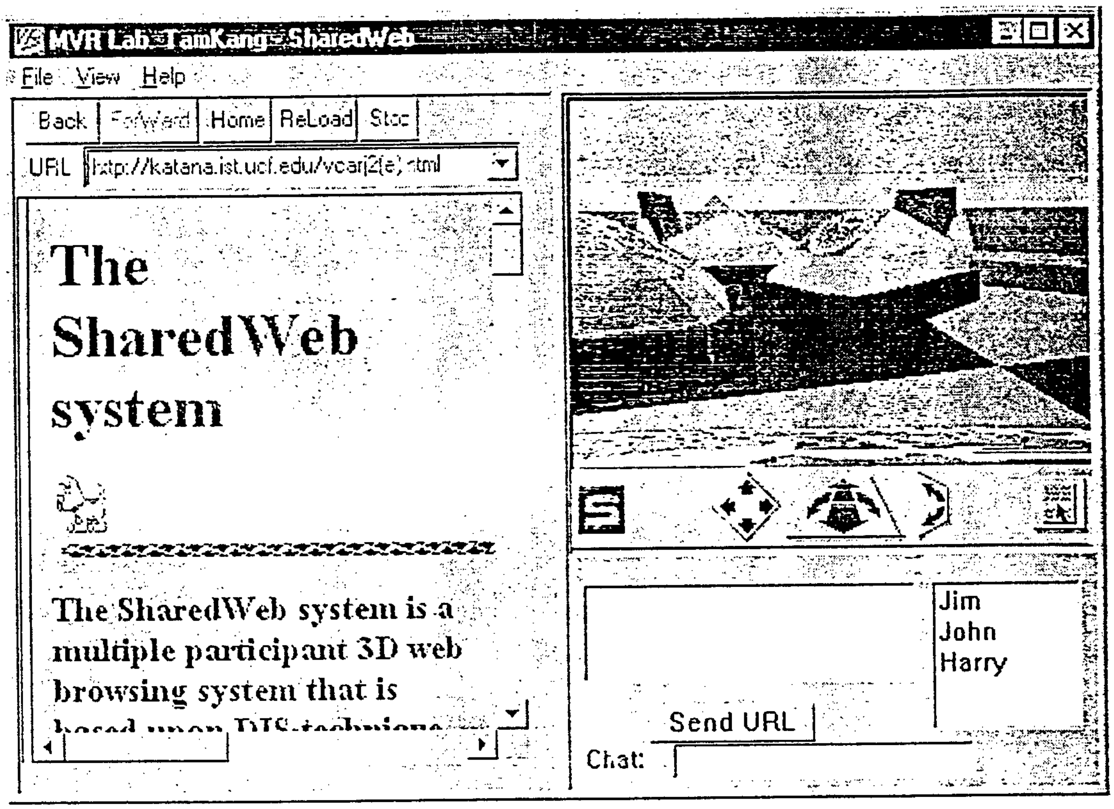


Figure 6. A scene in the Bumper Cars.

5.4 Tank Commander

This interactive tank combat game is a prototype a system integrating the 3D technique and WWW environment. This application is a 3D version of the role-playing game (RPG) simulation.

Traditionally, RPG is a two-dimensional graphical display environment. It has an interactive dialog box, or command sheet, for the user to control his character inside the game. Since the SharedWeb system fully integrates the HTML browser with the virtual reality navigator, the command sheet can be easily implemented by the HTML document so that the content of the command sheet can be dynamically modified. Also, a game master is required for RPG to control the game rules. For the SharedWeb system, the game master was implemented in the SharedWeb server.

The prototype system allows three users to enter virtual battlefield, the "bluetank," the "redtank," and the "referee". Bluetank and redtank can use the HTML document to control two tanks each in the virtual battlefield. Figure 3 in Section 4.4.2 shows the viewpoint of the red tank commander. The user command sheet has automatically downloaded into the HTML browser. The user can input data to control tank movements. In addition, check buttons allow the user to choose whether or not to fire tank missiles. After the user inputs the commands and presses the "Confirm" button, an HTTP PDU is sent to the Sharedweb server. The server then forwards it to all participants and the status of the tank is modified accordingly.

6. Conclusion and Future Works

The SharedWeb system has been successfully tested in both the Local Area Network (LAN) and Wide Area Network (WAN) environment. The major concern is the time delay in the information exchange among the SharedWeb browsers. The time delays in the LAN environment are very small. In WAN, however, the network congestion is always a concern. The SharedWeb system uses the Dead Reckoning technique to reduce the network traffic created by the interactions among the users. However, there is not much can be done to deal with the congestion in the Internet itself. After all, unlike DIS--which is running in an isolated network-- all the users in the world share the Internet. We have no control over when other users would create a heavy traffic in the Internet. In any case, the Dead Reckoning technique can maintain the interactivity even when the congested network delays the PDUs. After receiving a delayed PDU, proper delay compensation and smoothing are needed to make the interaction realistic. [31]

This report has presented the concept of seamless integration of the multi-user system into the WWW environment. With this seamless integration, the existing WWW server can easily support multi-user interactivity, and a shared virtual world can thus be easily constructed. The infrastructure and the communication protocol of this concept have been fully elaborated in this report. In addition, the architecture of a web-based multi-user system, called the SharedWeb system, has also been presented. Under the architecture of the SharedWeb system, the existing WWW server remains intact while supporting the multiple participant interactions. More significantly, the SharedWeb environment is transparent to the users. That is, each user only needs to download a scene file from a WWW server and logins with his nickname, and the WWW server will then gracefully transfer the user to an appropriate SharedWeb server.

Although the SharedWeb system successfully demonstrates the concept of seamless integration, more researches are still required to improve the multi-user interaction. For example, the consistency control of the shared objects is an important function for computer-supported cooperative work (CSCW) type of applications. [32] Moreover, in a virtual environment with N avatars, the current SharedWeb server must "broadcast" a change in any avatar's state to $N-1$ browsers. Thus, in the worst case, the server must send $N^2 - N$ PDUs to the network. This approach yields $O(N^2)$ complexity for updating messages and could exhaust the computation power of a host even before it saturates the network. To overcome this problem, the concept of spatial culling [33] is currently under investigation for the SharedWeb system.

Multi-server architecture of SharedWeb system is another important future work. This architecture can further scale up the number of participants as well as provide the fault tolerance feature on the server site.

References

[1] T. Berners-Lee. et al., "The World-Wide Web", *Communication of the ACM*, Vol. 37, No. 8, August 1994, pp. 76-82.

[2] "The Virtual Reality Modeling Language Specification Ver. 2.0", August 1996, available on line at <http://vrml.sgi.com/moving-worlds/spec>

[3] J. Y. Huang, J. L. Chang, C. W. Li and K. Lin, "Design of a Multiple Participant 3D War Game Environment over WWW", *Part of the SPIE Conference on Enabling Technology for Simulation Science II*, Orlando, Florida, April 1998, pp. 303-312.

[4] Y. Honda, K. Matsuda, J. Rekimoto, and R. Lea, "Virtual Society: Extending the WWW to Support a Multiuser Interactive shared 3D environment", *Procs. VRML'95*, San Diego, Calif., Aug 1995. available on line at <http://www.csl.sony.co.jp/person/rodger.html>

[5] W. Broll, "VRML : From the Web to Interactive Multi-user Virtual Reality", *Proc. Int'l Workshop MVD'95*, San Diego, Calif., Aug. 1995, available on line at <http://orgwis.gmd.de/~broll/papers/MVD95.ps.gz>

[6] O. Hagsand, "Interactive Multi-user Ves in the DIVE System", *IEEE Multimedia*, Vol. 3, No. 1, Spring 1996, pp. 30-39.

[7] "Community Place Browser", available on line at <http://www.sonypic.com/vs>

[8] R. C. Waters, et al., "Diamond Part and Spline: Social Virtual Reality with 3D Animation, Spoken Interaction, and Runtime Extendability", *Presence*, Vol. 6, No. 4, August 1997, pp. 461-481.

[9] R. C. Waters, D. B. Anderson, D. L. Schwenke, "The Interactive Sharing Transfer Protocol Version 1.0", *tech. Report*, Mitsubishi Electric Research Lab, Cambridge, Mass., 1997, available on line at <http://www.merl.com/reports/index.html/TR97-10>

[10] "Blaxxun Community Client", available on line at <http://ww3.blacksun.com/>

[11] "OnLive! Community Browser", available on line at <http://www.onlive.com>

[12] "OZ Virtual", available on line at <http://www.oz-inc.com/ov>

[13] "V*Realm Multi-User Browser", available on line at <http://www.ids-net.com/ids/vrealm>

[14] "Active Worlds", available on line at <http://www.activeworlds.com/>

[15] J. Y. Huang, C. T. Fang-Tson, J. C. Chen, and F. B. Wang, "SharedWeb -- A Shared Virtual Environment Over World Wide Web", *Pacific Graphics'97*, Korea, October 1997, pp. 178-189.

[16] J.Y. Huang, F.B. Wang, W.H. Hsu, and J.F. Chen, "Usage of DIS Technique to Create an Interactive WWW Environment", *14 th DIS Workshop*, Orlando, Florida, March 1996, pp. 201-210.

[17] C. Chee-Kai, "MOLTIP-UID and R-HTTP Specification", available on line at

<http://www.ontc.ncb.gov.sg/staff/cheekai/html/moltip-techspe.html>

[18] R. Sikorski and R. Peters, "A privacy Primer for the Web: Spam, Bread Crumbs, and Cookise", *Jama. The journal of the American medical association,* APR 1998 v279 n15 pp. 1219.

[19] R. Schwarz, "Manage Your Cookies With VB6 WebClasses", *Visual basic programmer's journal*, OCT 1998 v8 n12 pp. 88.

[20] IST, "Standard for Distributed Interactive Simulation--Application Protocols, Version 2.0", *IST Report No IST-CR-94-50*, March, 1994.

[21] K. C. Lin, "Dead reckoning and Distributed Interactive Simulation", *Proc. of SPIE's International Symposium on Aerospace, CR58*, Orlando, FL, April 1995, pp. 16-36.

[22] K. C. Lin and D. Schab, "Study on the Network Load in Distributed Interactive Simulation", *AIAA Journal of Aircraft*, Vol. 32, No. 6, 1995, pp. 1392-1394.

[23] R. Lea, Y. Honda, and K. Matsuda, "Virtual Society : Collaboration in 3D spaces on the Internet", available on line at <http://www.csl.sony.co.jp/project/VS/index.html>

[24] "The Common Gateway Interface", available on line at <http://hoohoo.ncsa.uiuc.edu/cgi/>

[25] "The Network Time Protocol (NTP) Distribution", available on line at <http://www.eecis.udel.edu/~ntp/ntp_spool/html/>

[26] D. L. Mills, "Clock Discipline Algorithms for the Network Time Protocol Version 4", *Proc. DIMACS Workshop on Networks in Distributed Computing*, March 1997, available on line at <http://www.eecis.udel.edu/~mills>

[27] J. Y. Huang, C. T. Fang-Tsou and J. L. Chang, "A Multiuser 3D Web Browsing System", *IEEE Internet Computing*, SEP. & OCT. 1998, pp. 70-80.

[28] J. Y. Huang, C. T. Fang-Tson, J. L. Chang and A. J. Lee, "Modeling of Multiple User Virtual Reality System with Petri Net Technique", *National Computer Symposium 1997*. In Chinese, December 1997, pp. F-81-F-88.

[29] J. Oikarinen and D. Reed, "Internet Relay Chat Protocol", *Internet RFC*, No. 1459, May 1993.

[30] "VRT for Windows – User Guide", Superscape Inc., UK, 1996.

[31] K. C. Lin, M. Wang, J. Wang, D. Schab, "The Smoothing of Dead Reckoning Image in Distributed Interactive Simulation", *AIAA Journal of Aircraft*, Vol. 33, No. 2, 1996, pp. 450-452.

[32] "Guide to Computer Supported Co-operative Working CSCW", available on line at < http://dougal.derby.ac.uk/andree/cscw6.html>

[33] C. Greenhalgh and S. Benford, "Massive, A Collaborative Virtual Environment for Teleconference", *ACM Trans. On Computer-Human Interaction*, Vol. 2, No. 3, 1995. pp. 239-261.

JOINT DOMAIN SPACE-TIME ADAPTIVE PROCESSING WITH SMALL TRAINING DATA SETS

Dimitris A. Pados
Assistant Professor
Department of Electrical Engineering
E-mail: pados@eng.buffalo.edu

The State University of New York at Buffalo
201 Bell Hall
Buffalo, NY 14260

Final Report for:
Summer Research Extension Program
Rome Laboratory

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

Rome Laboratory

and

State University of New York at Buffalo

March 1999

JOINT DOMAIN SPACE-TIME ADAPTIVE PROCESSING WITH SMALL TRAINING DATA SETS

Dimitris A. Pados
Assistant Professor
Department of Electrical Engineering
State University of New York at Buffalo
E-mail: pados@eng.buffalo.edu

Abstract

The classical problem of optimum detection of a complex signal of unknown amplitude in colored Gaussian noise is revisited. The focus, however, is on adaptive system designs with limited training data sets and low computational optimization complexity. In this context, the target vector is equipped with a carefully selected orthogonal auxiliary vector for disturbance suppression with one complex space-time degree of freedom. Direct generalization leads to adaptive generation of a sequence of conditionally optimized weighted auxiliary vectors that are orthogonal to each other and to the target vector of interest. This approach appears here for the first time. Adaptive disturbance suppression with any desired number of complex degrees of freedom below the data dimension is therefore possible. It is shown that processing with multiple auxiliary vectors falls under well known blocking-matrix processing principles. The proposed blocking matrix, however, is data dependent, adaptively generated, and no data eigen analysis is involved.

While the issues treated refer to general adaptive detection procedures, the presentation is given in the context of joint space-time adaptive processing for array radars.

JOINT DOMAIN SPACE-TIME ADAPTIVE PROCESSING

WITH SMALL TRAINING DATA SETS

Dimitris A. Pados

I. Introduction

Detection of a signal in Gaussian noise falls under the binary hypothesis testing version of the general Gaussian problem and lies in the foundation of the theory of optimum detection. Optimum data processing in the Bayes, Neyman-Pearson, maximum output Signal-to-Noise-Ratio (SNR), minimum Mean-Square-Error (MSE), or Minimum-Variance-Distortionless-Response (MVDR) sense are all well known, and lead to scaled versions of the same linear filter solution [1], [2], traditionally called Wiener or colored noise Matched Filter (MF). Calculation of this filter requires knowledge of the inverse covariance matrix of the noise process.

In the context of an array radar application with M antenna elements (spatial channels) and N pulses per coherent processing interval (CPI), optimum signal detection in the presence of spatial and temporal Gaussian noise requires joint space-time matched filtering in the MN complex vector space [3]. Since prior knowledge of the noise covariance matrix is not available, the maximum-likelihood sample average estimate is often used, developed from K noise only vector samples that correspond to distinct range cells. This is the so called secondary data set. Then, the inverse of the sample-matrix is considered as an estimate of the inverse covariance matrix. This approach is known as the Sample-Matrix-Inversion method (SMI) and in [4] it was shown to outperform the recursive least-mean-squares (LMS) adaptive implementation of the matched filter processor in terms of convergence rate and small-sample output SNR characteristics. Still, it was found that $K \geq 2MN$ independent and identically distributed (i.i.d.) training data samples are needed to maintain with probability 1/2 a loss lower than or equal to 3dB compared to the ideal MF. Simple variance normalization of the MF decision statistic led to a Constant False Alarm Rate (CFAR) test [5] for Neyman-Pearson detection.

System optimization in the generalized likelihood ratio (GLR) sense was pursued in [6]. The resulting test statistic offers embedded CFAR behavior, converges to the ideal MF solution at least in probability as the number of secondary data K grows, and it is shown in [5] to outperform the SMI approach for $K = 2MN$, except for high SNR regions. Extended performance analysis and studies of the sidelobe behavior of the GLR filter are carried out in [7].

It is important to note that the SMI and the GLR test are both asymptotically optimum to the extent that they converge in a probabilistic sense to the optimum ideal MF as the size of the secondary data set grows to infinity. However, for finite sample support no optimality can be claimed in either case and superior probability of detection performance for a fixed false alarm rate is theoretically possible by other filtering means. Moreover, both methods share the need to invert the sample covariance matrix of the noise process which implies: *(i)* general computational complexity of order $(MN)^3$, *(ii)* secondary data set of at least $K = 2MN$ samples to maintain with probability 1/2 a loss compared to the optimum rule of no more than 3dB, and *(iii)* $K \geq MN$ to ensure nonsingularity of the sample covariance matrix with probability 1 [6]. Property (ii) is due to characteristics of the Wishart distribution that the sample covariance matrix is assumed to follow under the i.i.d. and Gaussian assumptions. Property (iii) results from linear algebraic considerations of basis vectors needed to represent any MN-tuple vector.

Arguably, in airborne surveillance systems the training data size requirements make the practicality of these approaches questionable even for moderate values of M and N. This is particularly true if we consider a highly non-stationary, non-homogeneous [8] operating environment, typically encountered in practice. In this context, it is tempting to consider abolishing the concept of joint space-time processing and pursue separate ("disjoint" or "cascade") SMI or GLR detection in the M-dimensional space and the N-dimensional time domain. It is easy, however, to identify realistic cases where the performance of these cascade schemes falls unacceptably below that of the joint domain approaches [9].

In this present work the principle of joint space-time (S-T) processing is preserved but instead of seeking an adaptive implementation of the optimum linear filter in the C^{MN} complex space, suboptimum simpler solutions are pursued. The objective is linear space-time adaptive processing with low computational optimization complexity and sample support near or below the space-time product MN. In this context, the target vector of interest is accompanied by a carefully designed, data dependent, orthogonal auxiliary vector for joint S-T processing with *a single* complex degree of freedom. This approach is inductively extended to generate a sequence of conditionally optimized auxiliary vectors -orthogonal to each other and to the target of interest- for adaptive S-T processing with any desired number of complex degrees of freedom less than or equal to $MN - 1$. Interestingly enough, we show that processing with multiple auxiliary vectors falls under general blocking-matrix processing principles [10]-[14]. The proposed blocking matrix, however, is data dependent and no data eigen analysis is involved. Numerical and simulation results included in this work support the theoretical arguments and promote the new method as a desirable low computational cost alternative for joint space-time adaptive processing when very small secondary data training sets are available.

The SMI and GLR tests are treated as performance (probability of detection) and computational complexity benchmarks.

The rest of this manuscript is organized as follows. Section II introduces the system and signal models. The principles of Auxiliary-Vector filtering and the new test statistics are developed and studied in Section III. Probability of detection comparisons with the ideal Matched-Filter, the adaptive Matched-Filter and the Generalized-Likelihood-Ratio test are carried out in Section IV for a few representative cases.

II. Signal Model and Background

Consider a narrowband uniform linear radar array with M antenna elements (subarrays or spatial channels). Also, assume that each element collects the complex (I/Q) return of a series of N coherent pulses for some given range cell $k = 1, \ldots, K_{max} = T_{PRI}/T$, where T_{PRI} is the pulse repetition interval and T is the pulse duration. We organize the received data in the form of a matrix $X_{M \times N}$, where $X(m,n)$, $m = 1, \ldots, M$, $n = 1, \ldots, N$, denotes the m-element, n-pulse signal sample. The objective is to cope with system and surrounding disturbances and detect in $X_{M \times N}$ the presence of a desired signal of unknown amplitude. Without loss of generality and for notational simplicity, we consider a "vectorized" form of $X_{M \times N}$, where $Vec(X_{M \times N}) = \mathbf{X}_{MN \times 1}$ is constructed by sequencing all matrix columns in the form of a vector. From now on, bold variables indicate vectors in the C^{MN} complex space unless otherwise specified.

We begin by casting the detection problem in the context of binary hypothesis testing. We denote the disturbance only hypothesis by H_0 and the target plus disturbance hypothesis by

H_1.

$$\begin{aligned}\mathrm{H}_0 : \mathbf{X} &= \mathbf{J} + \mathbf{C} + \mathbf{N} \\ \mathrm{H}_1 : \mathbf{X} &= \alpha\mathbf{V} + \mathbf{J} + \mathbf{C} + \mathbf{N}. \end{aligned} \tag{1}$$

In (1), $\mathbf{J}$ represents a mixture of L broadband directional interferers (jammers) in the far field, where $\mathbf{J} = Vec(J_{M\times N})$ and

$$J(m,n) = \sum_{l=1}^{L} J_l(n)\, e^{j2\pi(m-1)\frac{sin\theta_l d}{\lambda}}, \tag{2}$$
$$n = 1,\ldots,N,\ m = 1,\ldots,M.$$

We assume that $J_l(n), l = 1,\ldots,L$, is complex white Gaussian distributed to account for channel fading at the pulse-repetition frequency as in the model of [15] and [16]. The antenna element spacing is d and the radar carrier wavelength is λ. The direction of arrival (DOA) $\theta_l, l = 1,\ldots,L$, is assumed to be uniformly distributed in $[-\pi/2, \pi/2]$. We find it convenient to define the spatial frequency

$$f_l \triangleq \frac{sin\theta_l d}{\lambda},\ l = 1,\ldots,L, \tag{3}$$

and assume that f_l is uniformly distributed in [-0.5,0.5] with proper selection of d and λ. In addition, $\mathbf{C}_{MN\times 1}$ in (1) accounts for colored Gaussian noise with covariance matrix $\mathbf{R}_c$ and corresponds to a radar clutter region. Spatially and temporally white disturbances are denoted by $\mathbf{N}$. The signal or "steering vector" of interest, $\mathbf{V}$, is present in $\mathbf{X}$ under hypothesis H_1 only. If $\mathbf{V}^H$ denotes the Hermitian operation on $\mathbf{V}$, then without loss of generality we assume that $\mathbf{V}^H\mathbf{V} = 1$ such that all energy signal characteristics are absorbed in the unknown complex amplitude constant α. For completeness, if $\mathbf{V} = Vec(V_{M\times N})$ then

$$V(m,n) = \frac{1}{\sqrt{MN}} e^{j2\pi(m-1)\frac{sin\theta_s d}{\lambda} + j2\pi(n-1)\frac{2v}{\lambda f_{PR}}}. \tag{4}$$

In (4), $\theta_s \in [-\pi/2, \pi/2]$ is the angle of arrival of the signal (target) of interest, ν is the target radial velocity, and f_{PR} is the radar pulse repetition frequency. Once again, it is convenient to define the spatial target frequency

$$f_s \triangleq \frac{sin\theta_s d}{\lambda} \tag{5}$$

and the "normalized Doppler" target frequency

$$f_D \triangleq \frac{2\nu}{\lambda f_{PR}}. \tag{6}$$

In this case we assume that $f_s, f_D \in [-0.5, 0.5]$.

Given a data vector $\mathbf{X}$ that corresponds to some range cell $k \in \{1, \ldots, K_{max}\}$, the objective is to decide in favor of H_0 or H_1 in a way that maximizes the probability of detection $P_D = Pr\{H_1\ decided|H_1\ true\}$ subject to a given false alarm constraint $P_{FA} = Pr\{H_1\ decided|H_0\ true\} \leq \rho$. The optimum decision rule is well known [1], [3] and of the form

$$|\mathbf{w}^H\mathbf{X}| \underset{H_0}{\overset{H_1}{\gtrless}} \tau, \tag{7}$$

where $\tau > 0$ is the threshold parameter to be determined according to the condition $P_{FA} = \rho$ and $\mathbf{w}$ is the linear filter defined by

$$\mathbf{w} = b\mathbf{R}^{-1}\mathbf{V}, \tag{8}$$

where b is an arbitrary positive scalar. In (8), $\mathbf{R} = E_{H_0}\{\mathbf{X}\mathbf{X}^H\}$ where $E_{H_i}\{\cdot\}$ denotes the statistical expectation operation under $H_i, i = 0, 1$. Then the variance under H_0 of the test statistic $\mathbf{w}^H\mathbf{X}$ is $Var_{H_0}\{\mathbf{w}^H\mathbf{X}\} = \mathbf{V}^H\mathbf{R}^{-1}\mathbf{V}$ which implies that the modified test statistic $\frac{\mathbf{V}^H\mathbf{R}^{-1}\mathbf{X}}{\sqrt{\mathbf{V}^H\mathbf{R}^{-1}\mathbf{V}}}$ is zero-mean unit-variance complex Gaussian distributed. As a consequence $\left|\frac{\mathbf{V}^H\mathbf{R}^{-1}\mathbf{X}}{\sqrt{\mathbf{V}^H\mathbf{R}^{-1}\mathbf{V}}}\right|^2$ is chi-square distributed with two degrees of freedom and leads to the familiar CFAR (constant false alarm rate) optimum decision rule

$$\frac{|\mathbf{V}^H\mathbf{R}^{-1}\mathbf{X}|^2}{\mathbf{V}^H\mathbf{R}^{-1}\mathbf{V}} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda, \qquad (\lambda > 0). \tag{9}$$

Substitution in (9) of the sample-average covariance matrix estimate $\hat{\mathbf{R}} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{X}_k\mathbf{X}_k^H$ from K data samples from H_0 defines the so called CFAR adaptive matched filter (AMF) detector [5]. In contrast, we recall that the generalized likelihood ratio (GLR) test of [6] and [7] is

$$\frac{|\mathbf{V}^H\hat{\mathbf{R}}^{-1}\mathbf{X}|^2}{\mathbf{V}^H\hat{\mathbf{R}}^{-1}\mathbf{V}(1+\frac{1}{K}\mathbf{X}^H\hat{\mathbf{R}}^{-1}\mathbf{X})} \underset{H_0}{\overset{H_1}{\gtrless}} \gamma, \qquad (\gamma > 0). \tag{10}$$

In the following section we develop new decision statistics that maintain the principle of linear filtering for signal detection in Gaussian disturbance. Compared to the AMF and GLR approaches we seek computational savings and superior performance for optimization with small secondary data sets ($K \leq MN$).

III. Adaptive Auxiliary-Vector Detection

To build progressively a new decision statistic we begin with the introduction of an arbitrary fixed "auxiliary" vector $\mathbf{G}_1$ in C^{MN} that is orthonormal with respect to the normalized target vector of interest $\mathbf{V}$:

$$\mathbf{G}_1{}^H\mathbf{V} = 0 \; \mathit{and} \; \mathbf{G}_1{}^H\mathbf{G}_1 = 1\,. \tag{11}$$

Next, we consider S-T adaptive processing with ***a single*** joint S-T degree of freedom in the form of $\mathbf{w}_{AV}^H\mathbf{X}$, where [17]

$$\mathbf{w}_{AV} \triangleq \mathbf{V} - \mu_1\mathbf{G}_1\,,\ \mu_1 \in C\,. \tag{12}$$

With an arbitrary but fixed auxiliary vector $\mathbf{G}_1$ that satisfies the constraints in (11), the filter in (12) can be optimized with respect to the complex scalar μ_1 only. The MS-optimum value of μ_1 can be identified from two different -yet equivalent- points of view. Since $\mathbf{w}_{AV}$ is by construction distortionless in the $\mathbf{V}$ direction ($\mathbf{w}_{AV}^H\mathbf{V} = 1$), we may seek the value of μ_1 that minimizes the output variance (energy) $E\{|\mathbf{w}_{AV}^H\mathbf{X}|^2\}$. An equivalent interpetation is to look for the value of μ_1 that minimizes the MSE $E\{(\mathbf{V}^H\mathbf{X} - \mu_1^*\mathbf{G}_1{}^H\mathbf{X})^2\}$ between the S-T main-beam processed data $\mathbf{V}^H\mathbf{X}$ and the orthogonal auxiliary-vector processed data $\mu_1^*\mathbf{G}_1^H\mathbf{X}$. The solution to the first minimum output variance problem can be obtained by direct differentiation of the variance expression. The solution to the second MSE problem can be found in the context of linear MS-optimum estimation [2]. The solution is, of course, identical in both cases and it is identified by the following proposition.

Proposition 1 *The value of the complex scalar μ_1 that minimizes the output variance of the filter $\mathbf{w}_{AV} \triangleq \mathbf{V} - \mu_1\mathbf{G}_1$ or minimizes the mean square error between the main-beam processed data $\mathbf{V}^H\mathbf{X}$ and the auxiliary-vector processed data $\mu_1^*\mathbf{G}_1^H\mathbf{X}$ is*

$$\mu_1 = \frac{\mathbf{G}_1^H\mathbf{R}\mathbf{V}}{\mathbf{G}_1^H\mathbf{R}\mathbf{G}_1}. \quad \square \tag{13}$$

Proposition 1 identifies the optimum value of μ_1 for *any* fixed auxiliary vector $\mathbf{G}_1$. We now turn our attention to the selection of an auxiliary vector subject to an appropriately chosen criterion. We understand that there exists a vector $\mathbf{G}_1$ in the subspace orthogonal to $\mathbf{V}$ for which $\mathbf{w}_{AV}$ and the optimum filter $\mathbf{w}$ in (8) coincide. This solution, however, is not desirable since it reverts to processing with full MN degrees of freedom and requires inversion of the pertinent S-T covariance matrix. The selection criterion for the auxiliary vector $\mathbf{G}_1$ that we propose in this work is motivated by the MSE interpretation of the filter in (12) and it is the maximization of the magnitude of the crosscorrelation between the $\mathbf{V}$ processed data $\mathbf{V}^H\mathbf{X}$ and the auxiliary-vector processed data $\mathbf{G}_1^H\mathbf{X}$, subject to the constraints in (11):

$$\mathbf{G}_1 = \arg\max_{\mathbf{G}_1} |E\{\mathbf{V}^H\mathbf{X}(\mathbf{G}_1^H\mathbf{X})^*\}| = \arg\max_{\mathbf{G}_1} |\mathbf{V}^H\mathbf{R}\mathbf{G}_1|, \tag{14}$$
$$\text{subject to } \mathbf{G}_1^H\mathbf{V} = 0 \text{ and } \mathbf{G}_1^H\mathbf{G}_1 = 1\,.$$

We note that both the criterion function $|\mathbf{V}^H\mathbf{R}\mathbf{G}_1|$ and the constraints are phase invariant. In other words, if $\mathbf{G}_1'$ maximizes $|\mathbf{V}^H\mathbf{R}\mathbf{G}_1|$ under the given constraints, so does $\mathbf{G}_1'e^{-j\phi}$ for any phase ϕ. Therefore, without loss of generality, we may identify the unique auxiliary vector that is a solution to the constraint optimization problem in (20) and places $\mathbf{V}^H\mathbf{R}\mathbf{G}_1$ on the non-negative real line ($\mathbf{V}^H\mathbf{R}\mathbf{G}_1 \geq 0$). The following proposition identifies the optimum auxiliary vector according to our criterion. The proof is omitted due to lack of space.

Proposition 2 *The auxiliary vector*

$$\mathbf{G}_1 = \frac{\mathbf{R}\mathbf{V} - (\mathbf{V}^H\mathbf{R}\mathbf{V})\mathbf{V}}{\|\mathbf{R}\mathbf{V} - (\mathbf{V}^H\mathbf{R}\mathbf{V})\mathbf{V}\|} \tag{15}$$

maximizes the magnitude of the crosscorrelation between the main-beam processed data $\mathbf{V}^H\mathbf{X}$ (w.l.o.g. $\mathbf{V}^H\mathbf{V} = 1$) and the auxiliary vector processed data $\mathbf{G}_1^H\mathbf{X}$, $|\mathbf{V}^H\mathbf{R}\mathbf{G}_1|$, subject to the constraints $\mathbf{G}_1^H\mathbf{V} = 0$ and $\mathbf{G}_1^H\mathbf{G}_1 = 1$. $\square$

Propositions 1 and 2 identify completely the S-T adaptive filter in (12). To generalize these results, we may consider a filter $\mathbf{w}_c$ consisting of P, $1 \leq P \leq MN - 1$, weighted auxiliary vectors that are orthonormal with respect to each other and to the normalized target vector $\mathbf{V}$:

$$\mathbf{w}_c \triangleq \mathbf{V} - \sum_{i=1}^{P} \mu_i \mathbf{G}_i. \tag{16}$$

The sequence of auxiliary vectors $\mathbf{G}_i$ and weights μ_i, $i = 1, \ldots, P$, are conditionally optimized and defined inductively. First, $\mathbf{G}_1$ and μ_1 are optimized exactly as in (15) and (13). Next, we upgrade the "main-beam" to $\mathbf{V} - \mu_1\mathbf{G}_1$ and we seek an auxiliary vector $\mathbf{G}_2$ that maximizes the crosscorrelation magnitude $|(\mathbf{V} - \mu_1 \mathbf{G}_1)^H \mathbf{R} \mathbf{G}_2|$ subject to the orthonormality constraints $\mathbf{G}_2^H \mathbf{V} = 0$, $\mathbf{G}_2^H \mathbf{G}_1 = 0$, and $\mathbf{G}_2^H \mathbf{G}_2 = 1$. Arguing as in Proposition 2 we find

$$\mathbf{G}_2 = \frac{\mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1) - [\mathbf{V}^H \mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1)]\mathbf{V} - [\mathbf{G}_1^H \mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1)]\mathbf{G}_1}{\|\mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1) - [\mathbf{V}^H \mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1)]\mathbf{V} - [\mathbf{G}_1^H \mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1)]\mathbf{G}_1\|}. \tag{17}$$

The conditionally MS-optimum value of μ_2 given $\mathbf{G}_1$, μ_1, and $\mathbf{G}_2$ is

$$\mu_2 = \frac{\mathbf{G}_2^H \mathbf{R}(\mathbf{V} - \mu_1 \mathbf{G}_1)}{\mathbf{G}_2^H \mathbf{R} \mathbf{G}_2}. \tag{18}$$

To complete the recursive construction of the filter $\mathbf{w}_c$ in (16) let us assume that $\mathbf{G}_i$, μ_i are defined for $i = 1, \ldots, p$, $1 \leq p \leq P-1$. Then, with "main-beam" $\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i$, the auxiliary vector $\mathbf{G}_{p+1}$ that maximizes the crosscorrelation magnitude $|(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)^H \mathbf{R} \mathbf{G}_{p+1}|$ subject to the orthonormality constraints is

$$\mathbf{G}_{p+1} = \frac{\mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i) - [\mathbf{V}^H \mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)]\mathbf{V} - \sum_{j=1}^{p} [\mathbf{G}_j^H \mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)]\mathbf{G}_j}{\|\mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i) - [\mathbf{V}^H \mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)]\mathbf{V} - \sum_{j=1}^{p} [\mathbf{G}_j^H \mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)]\mathbf{G}_j\|}. \tag{19}$$

The conditionally MS-optimum coefficient μ_{p+1} becomes

$$\mu_{p+1} = \frac{\mathbf{G}_{p+1}^H \mathbf{R}(\mathbf{V} - \sum_{i=1}^{p} \mu_i \mathbf{G}_i)}{\mathbf{G}_{p+1}^H \mathbf{R} \mathbf{G}_{p+1}}. \tag{20}$$

Equations (19) and (20) create a sequence of conditionally optimized auxiliary vectors and conditionally optimized weights and identify completely the filter under consideration $\mathbf{w}_c$ in (16). To seek further improvements in the form of unconditional weight optimization and to relate with existing literature, we find it convenient to organize the auxiliary vectors in the form of an $MN \times P$ matrix $\mathbf{B}$ where $\mathbf{G}_p$, $p = 1, \ldots, P$, constitutes the p-th column of $\mathbf{B}$:

$$\mathbf{B}_{MN \times P} = [\mathbf{G}_1 \, \mathbf{G}_2 \, \ldots \, \mathbf{G}_P]. \tag{21}$$

With this setup the orthonormality conditions translate to

$$\mathbf{B}^H \mathbf{V} = \mathbf{0}_{P \times 1} \;\; \textit{and} \;\; \mathbf{B}^H \mathbf{B} = \mathbf{1}_{P \times P}, \tag{22}$$

and the overall linear filter $\mathbf{w}_c$ in (16) can be written compactly as

$$\mathbf{w}_c \triangleq \mathbf{V} - \sum_{i=1}^{P} \mu_i \mathbf{G}_i = \mathbf{V} - \mathbf{B}\boldsymbol{\mu}. \tag{23}$$

We call $\mathbf{B}$ a *"blocking"* matrix because it blocks signal components in the S-T direction of interest $\mathbf{V}$. Blocking matrix processing techniques have been of considerable interest in array processing applications in the past [10]. In fact, some recent research efforts focus on the selection of the blocking processor based on eigenvector metrics [11]-[12] or data independent Discrete-Fourier (DFT) or Walsh-Hadamard transforms (WHT) [18]. In this present work, the formulation of the blocking matrix $\mathbf{B}$ in (21) is data dependent with low computational complexity and no eigen analysis involved. With respect to the conditionally optimized weights in the form of the vector $\boldsymbol{\mu}$, an alternative filter design that involves the unconditional MS-optimum vector weight value $\boldsymbol{\mu}^o$ can be sought:

$$\mathbf{w}_{\mathbf{B}} \triangleq \mathbf{V} - \mathbf{B}\boldsymbol{\mu}^o. \tag{24}$$

There are two equivalent interpretations for the MS-optimum value of the weight vector $\boldsymbol{\mu}^o$. Since the overall blocking matrix filter $\mathbf{w}_{\mathbf{B}}$ in (24) is by construction distortionless in the $\mathbf{V}$ direction of interest ($\mathbf{w}_{\mathbf{B}}{}^H\mathbf{V} = 1$), we may search for the weight vector $\boldsymbol{\mu}^o$ that minimizes the output variance (energy) $E\{|\mathbf{w}_{\mathbf{B}}{}^H\mathbf{X}|^2\}$. Equivalently, we may instead identify the vector $\boldsymbol{\mu}^o$ that minimizes the MSE $E\{(\mathbf{V}^H\mathbf{X} - \boldsymbol{\mu}^{oH}\mathbf{B}^H\mathbf{X})^2\}$ between the main-beam processed data $\mathbf{V}^H\mathbf{X}$ and the blocking-matrix processed data $\boldsymbol{\mu}^{oH}\mathbf{B}^H\mathbf{X}$. The solution to the first problem statement can be obtained by direct differentiation of the variance expression. The theory of linear MS-optimum estimation [2] offers the solution to the second problem. The solutions, of course, coincide and the optimum weight vector is of the form of a generalized sidelobe canceler identified by Proposition 3 below. This result is a generalization of Proposition 1.

Proposition 3 *The P-dimension, $1 \leq P \leq MN-1$, complex weight vector $\boldsymbol{\mu}^o$ that minimizes the output variance of the filter $\mathbf{w_B} \triangleq \mathbf{V} - \mathbf{B}\boldsymbol{\mu}^o$ or minimizes the mean square error between the main-beam processed data $\mathbf{V}^H\mathbf{X}$ and the blocking-matrix processed data $\boldsymbol{\mu}^{oH}\mathbf{B}^H\mathbf{X}$ is*

$$\boldsymbol{\mu}^o = [\mathbf{B}^H\mathbf{R}\mathbf{B}]^{-1}\mathbf{B}^H\mathbf{R}\mathbf{V}. \;\square \tag{25}$$

We conclude this section with a few observations. Blocking matrix processing with P joint S-T degrees of freedom requires inversion of the $P \times P$ blocked-data covariance matrix $\mathbf{B}^H\mathbf{R}\mathbf{B}$, as seen by (25). The output variance of the *ideal* filter $\mathbf{w_B}$ as a function of the blocking matrix $\mathbf{B}$ can be seen to be

$$VAR_{\mathbf{B}} \triangleq E\{|\mathbf{w_B}^H\mathbf{X}|^2\} = \mathbf{V}^H\mathbf{R}\mathbf{V} - \mathbf{V}^H\mathbf{R}\mathbf{B}[\mathbf{B}^H\mathbf{R}\mathbf{B}]^{-1}\mathbf{B}^H\mathbf{R}\mathbf{V}. \tag{26}$$

By construction, comparison with the output variance of the *ideal* MF filter in (8), $VAR_{MF} = \mathbf{V}^H\mathbf{R}^{-1}\mathbf{V}$, and the output variance of the *ideal* $\mathbf{w}_c$ filter in (16), VAR_c, shows that $VAR_{MF} \leq VAR_{\mathbf{B}} \leq VAR_c$ and $VAR_{MF} = VAR_{\mathbf{B}}$ if $P = MN-1$, while $VAR_{\mathbf{B}} = VAR_c$ if $P = 1$.

For the proposed filters $\mathbf{w}_c$ in (16) and $\mathbf{w_B}$ in (24) and any number of auxiliary vectors $1 \leq P \leq MN-1$, the decision statistics $\frac{\mathbf{w}_c^H\mathbf{X}}{\sqrt{\mathbf{w}_c^H\mathbf{R}\mathbf{w}_c}}$ and $\frac{\mathbf{w_B}^H\mathbf{X}}{\sqrt{\mathbf{w_B}^H\mathbf{R}\mathbf{w_B}}}$ are *0*–mean, unit-variance complex Gaussian. Then,

$$\frac{|\,\mathbf{w}_c^H\mathbf{X}\,|^2}{\mathbf{w}_c^H\mathbf{R}\mathbf{w}_c} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda \;\; and \;\; \frac{|\,\mathbf{w_B}^H\mathbf{X}\,|^2}{\mathbf{w_B}^H\mathbf{R}\mathbf{w_B}} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda \tag{27}$$

define CFAR tests in parallel to the optimum matched-filter test in (9). In adaptive implementations of $\mathbf{w}_c$ and $\mathbf{w_B}$, say $\hat{\mathbf{w}}_c$ and $\hat{\mathbf{w}}_{\mathbf{B}}$, the covariance matrix $\mathbf{R}$ needed in (19), (20), and (25) is estimated through the maximum likelihood sample-average procedure as usual:

$$\hat{\mathbf{R}} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{X}(k)\mathbf{X}(k)^H. \tag{28}$$

Then, the adaptive auxiliary-vector tests become

$$\frac{|\hat{\mathbf{w}}_c^H \mathbf{X}|^2}{\hat{\mathbf{w}}_c^H \hat{\mathbf{R}} \hat{\mathbf{w}}_c} \underset{H_0}{\overset{H_1}{\gtrless}} \hat{\lambda}_1 \quad and \quad \frac{|\hat{\mathbf{w}}_{\mathbf{B}}^H \mathbf{X}|^2}{\hat{\mathbf{w}}_{\mathbf{B}}^H \hat{\mathbf{R}} \hat{\mathbf{w}}_{\mathbf{B}}} \underset{H_0}{\overset{H_1}{\gtrless}} \hat{\lambda}_2. \tag{29}$$

The tests are asymptotically CFAR (as $K \rightarrow \infty$, $\hat{\lambda_1}, \hat{\lambda_2} \rightarrow \lambda$ in (9) for the optimum normalized matched-filter test for any given false alarm rate P_{FA}). For finite training sets they qualify as Cell-Averaging CFAR (CA-CFAR), as seen by the test denominators.

IV. Numerical and Simulation Comparisons

In this section we support and illustrate the preceding theoretical developments through four representative case-studies based on the hypothesis testing problem in (1). In all cases we assume presence of a mixture of 3 (three) broadband interferers with Jammer-to-Noise-Ratio $JNR = \frac{E\{|\mathbf{J}_l|^2\}}{\sigma^2} \approx 35dB$, $l = 1,2,3$. The jammers follow the model in (2) and the spatial frequency f_l, $l = 1,2,3$, is randomly drawn from the uniform $[-0.5, 0.5]$ distribution. The "peak" clutter (colored Gaussian noise) to noise ratio is fixed at $CNR = \frac{\sigma_c^2}{\sigma^2} = 40dB$. The false alarm rate is set at $P_{FA} = .01$ and for simulation purposes threshold and probability of detection estimates P_D are based on 10,000 samples from H_0 and H_1 respectively. All presented results are averages over 100 independent Monte-Carlo runs for arbitrarily chosen target vectors and jammers. P_D values are plotted as a function of the total SINR defined by

$$SINR = |\alpha|^2 \mathbf{V}^H \mathbf{R}^{-1} \mathbf{V}. \tag{30}$$

We assume a radar array system with $M = 5$ antenna elements (channels) and $N = 12$ pulses per coherent processing interval (CPI). All studies include the ideal matched filter as a reference point, and the adaptive matched filter, the GLR test, the adaptive $\mathbf{w}_c$ filter in (16) with one and ten auxiliary vectors ($P = 1$ and $P = 10$), and the adaptive $\mathbf{w}_\mathbf{B}$ filter in (24) with thirty auxiliary vectors ($P = 30$), all for secondary data sets of size $K = MN = 60$.

Figures 1 through 4 present probability of detection versus SINR results for four different clutter scenarios generated as in [19] with normalized one-lag temporal clutter correlation 0.4, 0.6, 0.97, and 0.999, respectively. The eigen profiles of the corresponding true clutter covariance matrices are shown in Figs. 5 through 8. We again note that the algorithmic developments in this work do not involve data eigen analysis. However, the eigen profiles provide an understanding of the need for an increased number of auxiliary vectors with unconditional weight optimization when operating in the presence of high temporal correlation clutter.

Relatively low temporal clutter correlation (low pulse-repetition frequency f_{PR} and/or high clutter velocity) results to benign eigenvalue distributions as seen in Figs. 5 and 6. In such cases the proposed filter $\mathbf{w}_c$ in (16) offers significant performance gains even with very few auxiliary vectors and, therefore, minimal computational cost.

For higher temporal clutter correlation (high pulse-repetition frequency f_{PR} and low clutter velocity) the proposed filter $\mathbf{w}_\mathbf{B}$ in (24) becomes increasingly more effective. For the somewhat typical case of one-lag temporal clutter correlation 0.999 in Fig. 4, performance gains in excess of $7dB$ compared to the GLR and adaptive MF counterparts can be seen. This is particularly true for high SINR values and detection with probability close to 1.

Acknowledgement

The authors would like to thank Dr. Muralidhar Rangaswamy for comments on the original manuscript.

References

[1] H. L. Van Trees, *Detection, Estimation and Modulation Theory, Part I* , New York, NY: Wiley, 1968.

[2] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms*, Englewood Cliffs, NJ: Prentice Hall, 1995.

[3] L. E. Brennan and I. S. Reed, "Theory of adaptive radar," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-9, pp. 237-252, Mar. 1973.

[4] L. S. Reed, J. D. Mallet, and L. E. Brennan, "Rapid convergence rate in adaptive arrays," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-10, pp. 853-863, Nov. 1974.

[5] F. C. Robey, D. R. Fuhrmann, E. J. Kelly, and R. Nitzberg, "A CFAR adaptive matched filter detector," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-28, pp. 208-216, Jun. 1992.

[6] E. J. Kelly, "An adaptive detection algorithm," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-22, pp. 115-127, Mar. 1986.

[7] E. J. Kelly, "Performance of an adaptive detection algorithm; Rejection of unwanted signals," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-25, pp. 122-133, Mar. 1989.

[8] W. L. Melvin and M. C. Wicks, "Improving practical space-time adaptive radar," in *Proc. of 1997 IEEE National Radar Conf.*, pp. 48-53, Syracuse, NY, May 1997.

[9] H. Wang and L. Cai, "On adaptive spatial-temporal processing for airborne surveillance radar systems," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-30, pp. 660-670, Jul. 1994.

[10] S. P. Applebaum, "Adaptive arrays," *IEEE Trans. Antennas Propagation*, vol. 24, pp. 585-598, Sept. 1976.

[11] B. D. Van Veen and R. A. Roberts, "Partially adaptive beamformer design via output power minimization," *IEEE Trans. on Acoust. Speech and Sign. Proc.*, vol. 35, pp. 1524-1532, Nov. 1987.

[12] J. S. Goldstein and I. S. Reed, "Reduced-rank adaptive filtering," *IEEE Trans. Signal Processing*, vol. 45, pp. 492-496, Feb. 1997.

[13] A. M. Haimovich, C. Peckham, T. Ayoub, J. S. Goldstein, and I. S. Reed, "Performance analysis of reduced-rank STAP," in *Proc. of 1997 IEEE National Radar Conf.*, pp. 42-47, Syracuse, NY, May 1997.

[14] A. M. Haimovich, "The eigencanceler: adaptive radar by eigenanalysis methods," *IEEE Trans. Aerospace and Electr. Syst.*, vol. AES-32, pp. 532-542, Apr. 1996.

[15] J. R. Roman and D. W. Davis, "Multichannel system identification and detection theory using output data techniques: Volume II", USAF Rome Laboratory, report No. SSC-TR-96-02, Oct. 1996.

[16] J. Ward, "Space-time adaptive processing for airborne radar", MIT Lincoln Laboratory, Technical Report No. TR-1015, Dec. 1994.

[17] D. A. Pados and S. N. Batalama, "Low complexity blind detection of DS/CDMA signals: Auxiliary-vector receivers," *IEEE Trans. Communications*, vol. 45, pp. 1586-1594, Dec. 1997.

[18] J. B. Schodorf and D. B. Williams, "Array processing techniques for multiuser detection," *IEEE Trans. Comm.*, vol. 45, pp. 1375-1378, Nov. 1997.

[19] J. H. Michels, T. Tsao, B. Himed, and M. Rangaswamy, "Space-time adaptive processing (STAP) in airborne radar applications," in *Proc. IASTED Intern. Conf. on Sign. Proc. and Comm.*, Canary Islands, Spain, Feb. 1998.

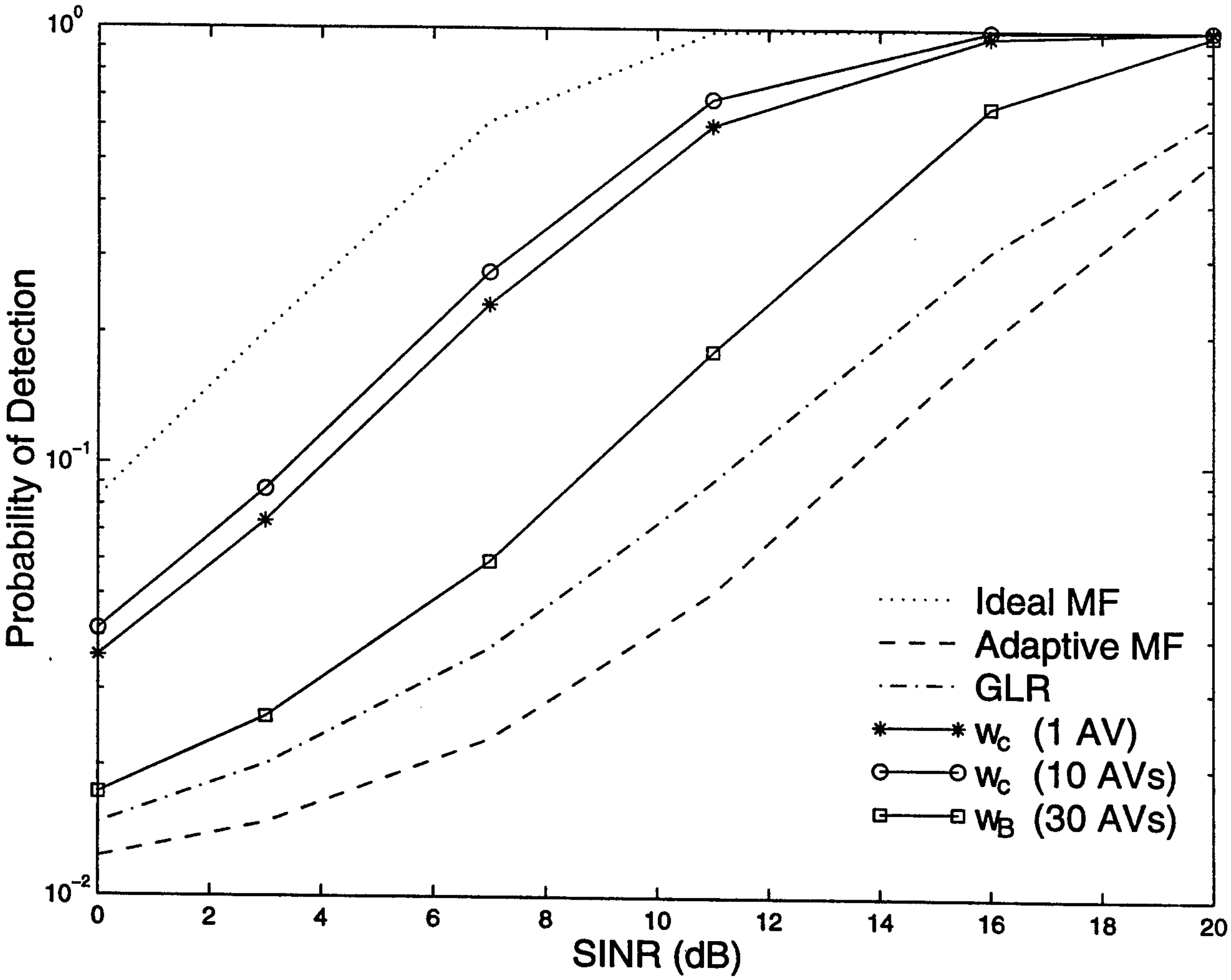


Figure 1: P_D vs SINR (one-lag temporal clutter correlation 0.4).

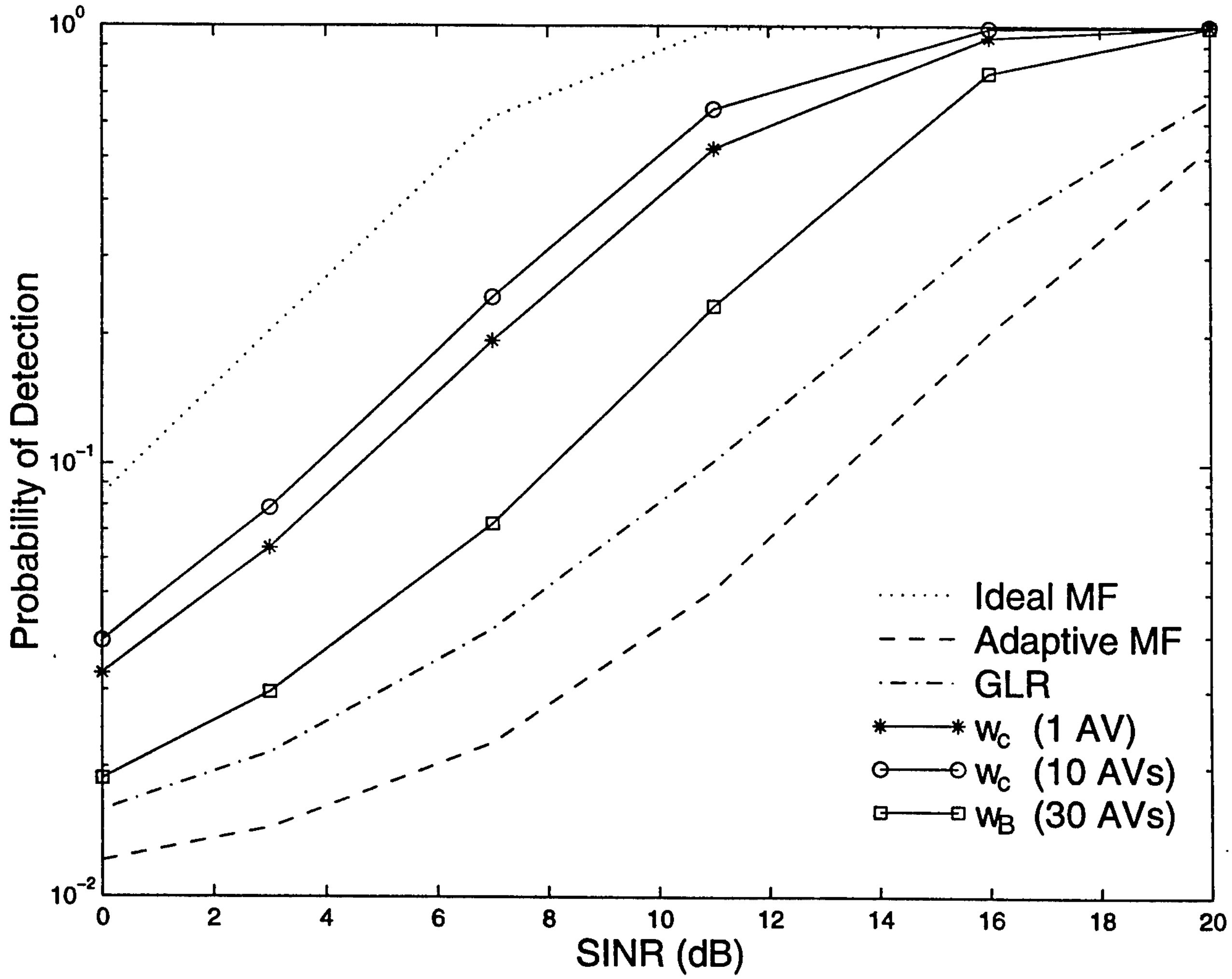


Figure 2: P_D vs SINR (one-lag temporal clutter correlation 0.6).

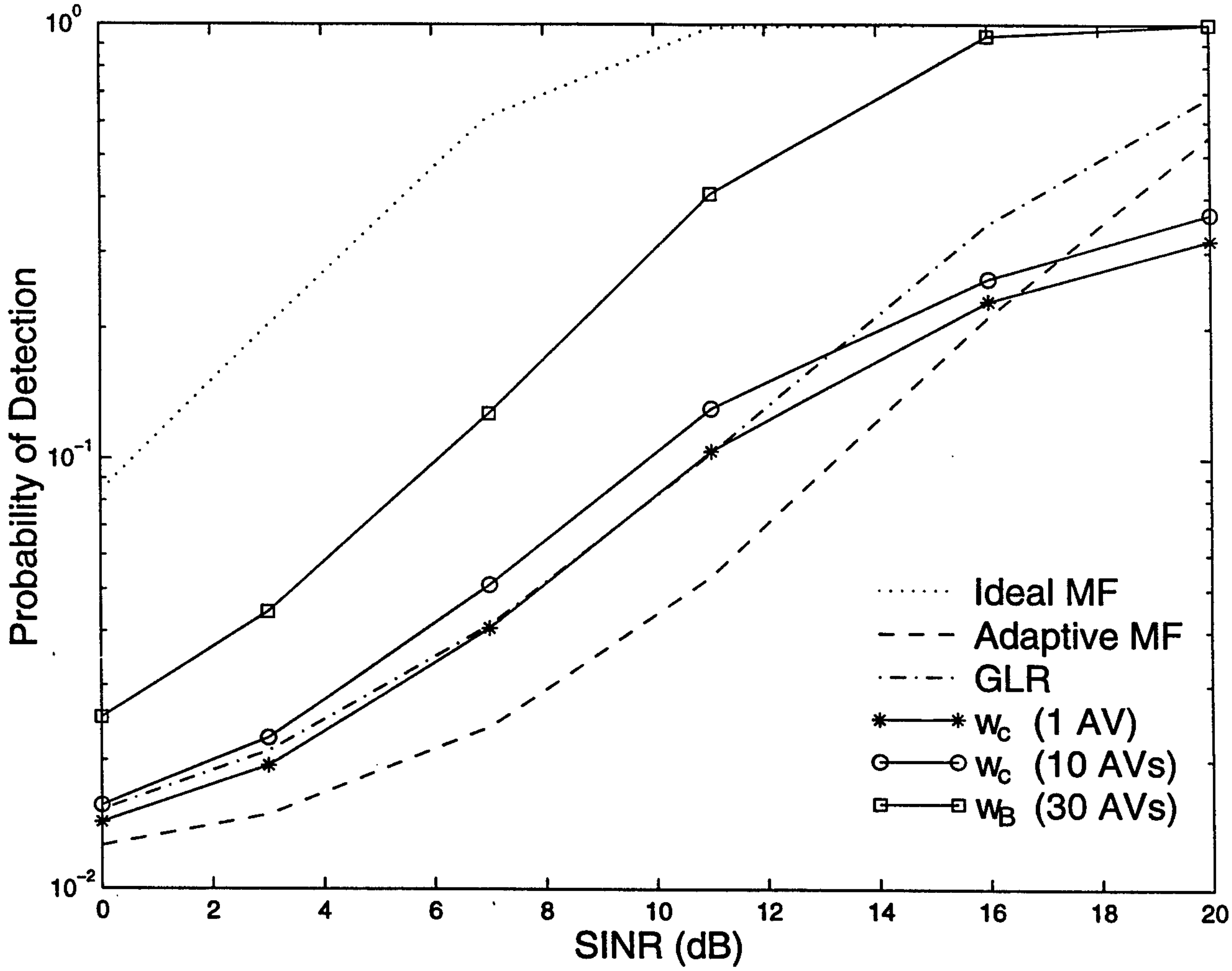


Figure 3: P_D vs SINR (one-lag temporal clutter correlation 0.97).

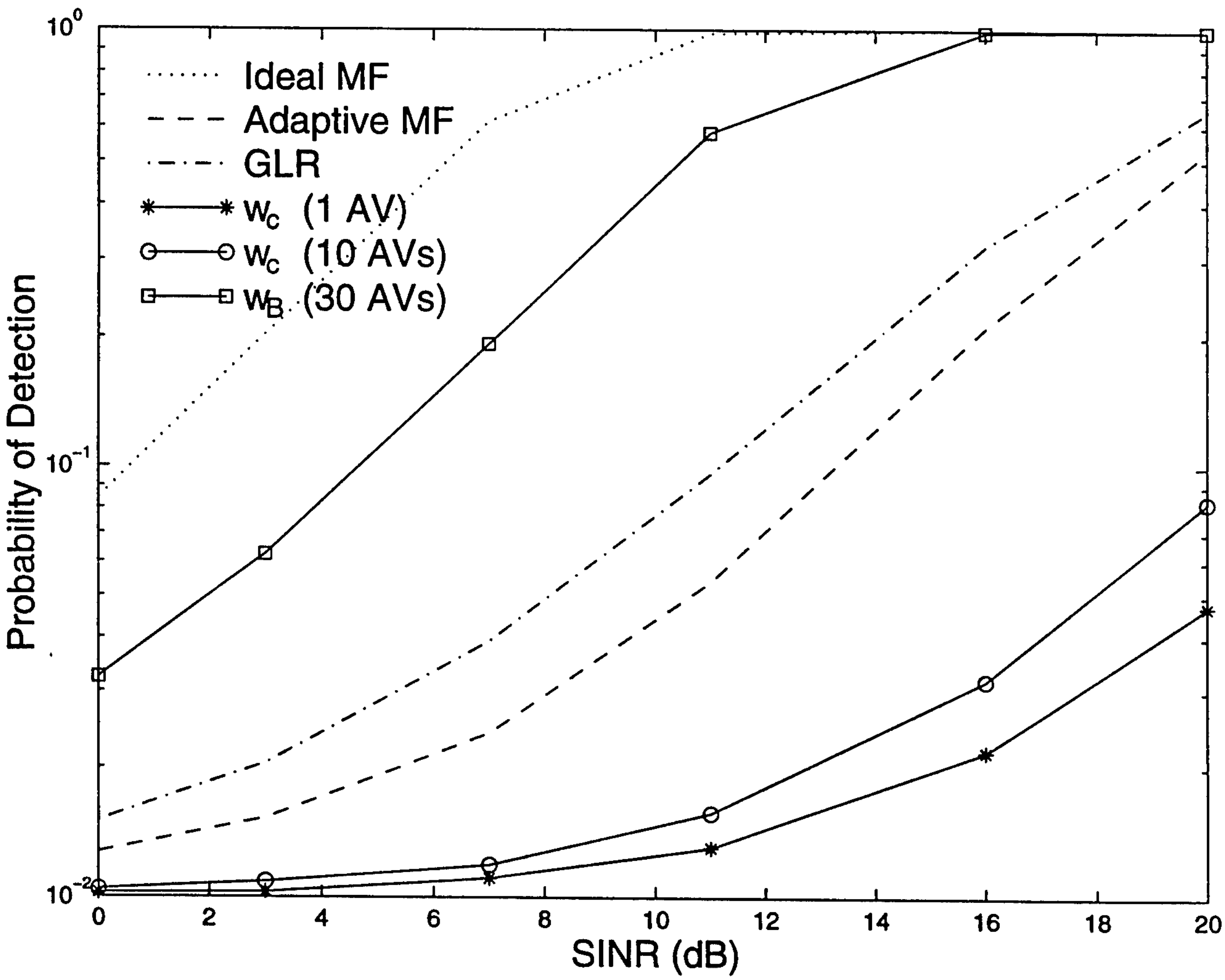


Figure 4: P_D vs SINR (one-lag temporal clutter correlation 0.999).

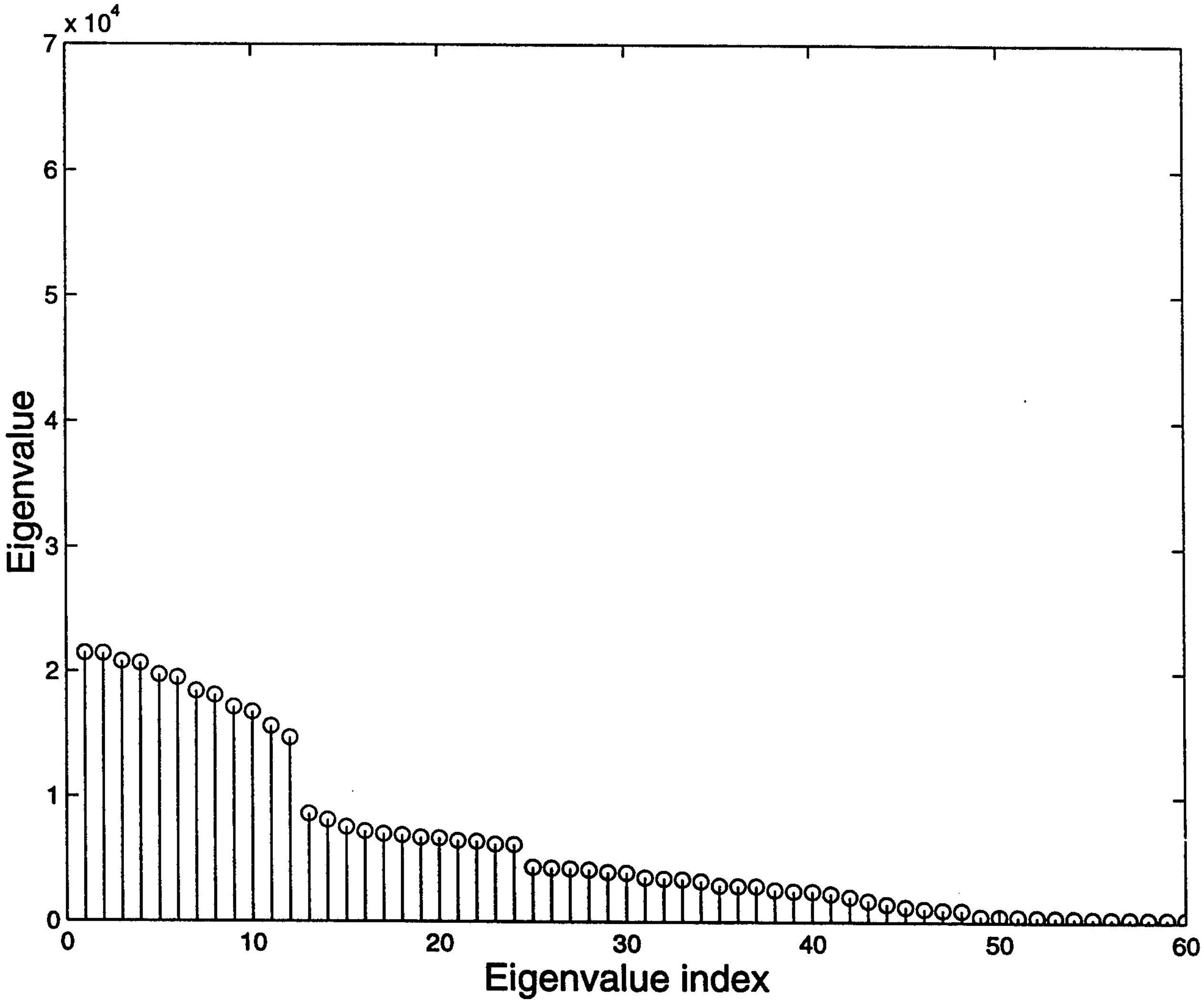


Figure 5: Eigen profile of clutter covariance matrix in Fig. 1.

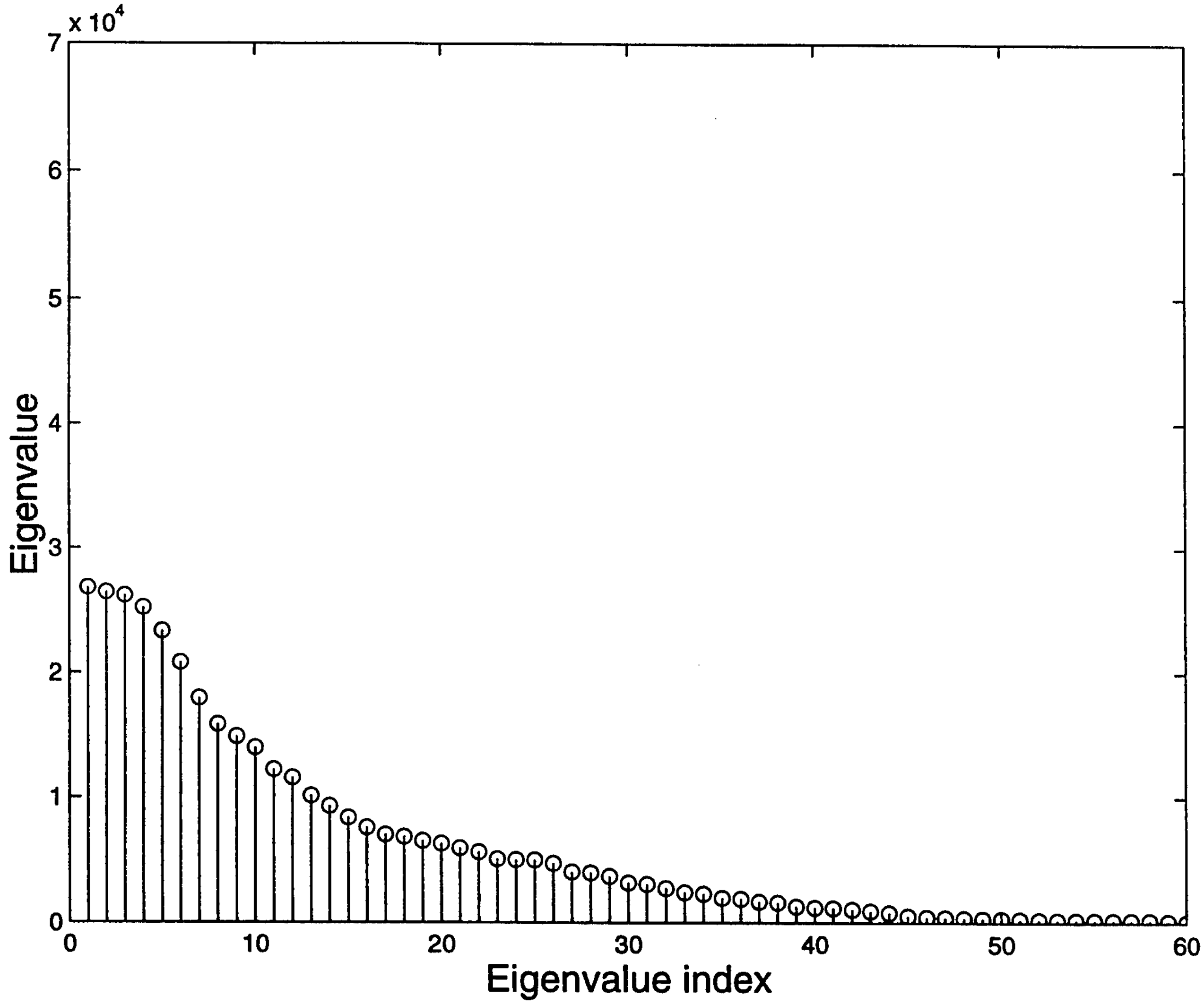


Figure 6: Eigen profile of clutter covariance matrix in Fig. 2.

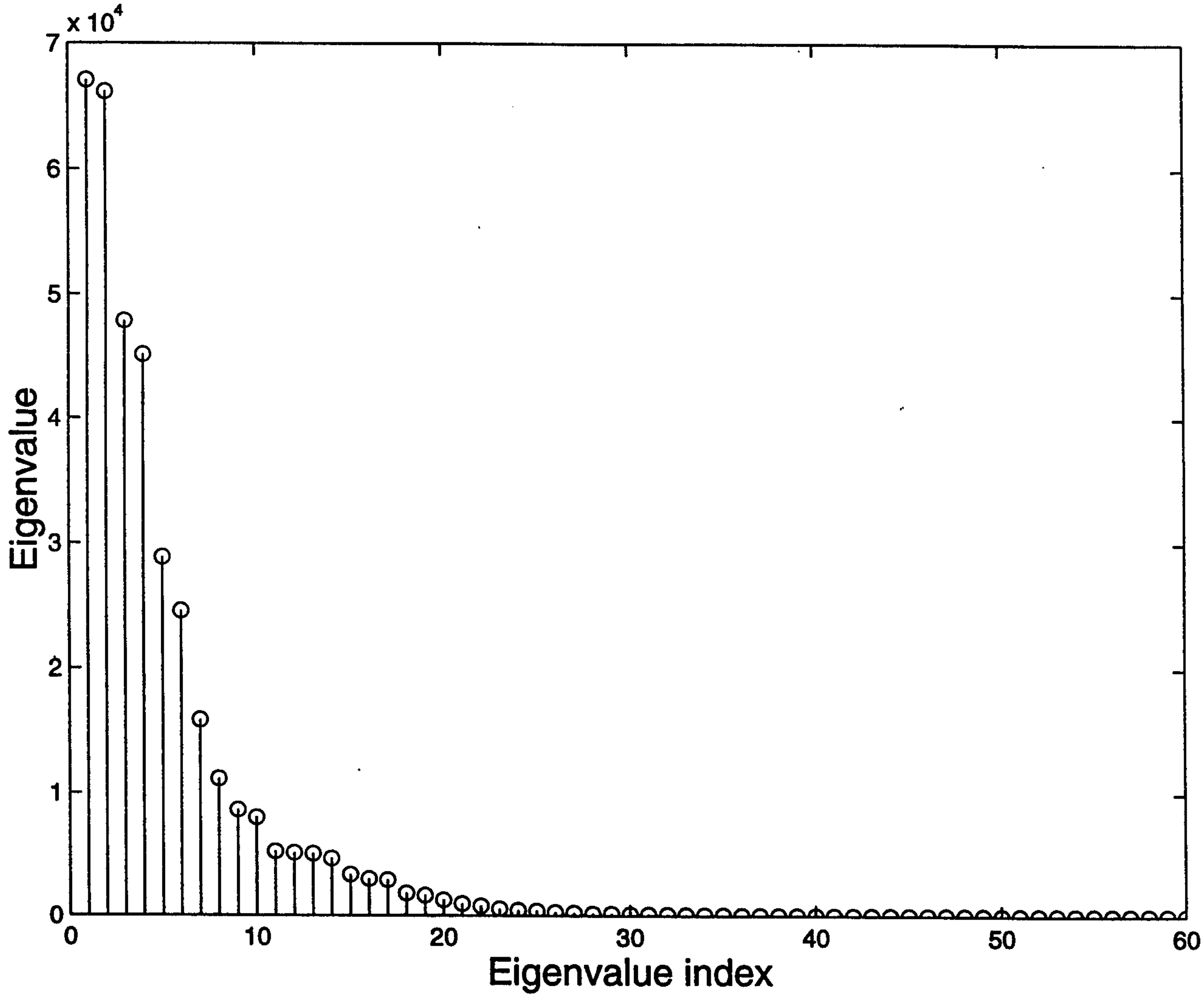


Figure 7: Eigen profile of clutter covariance matrix in Fig. 3.

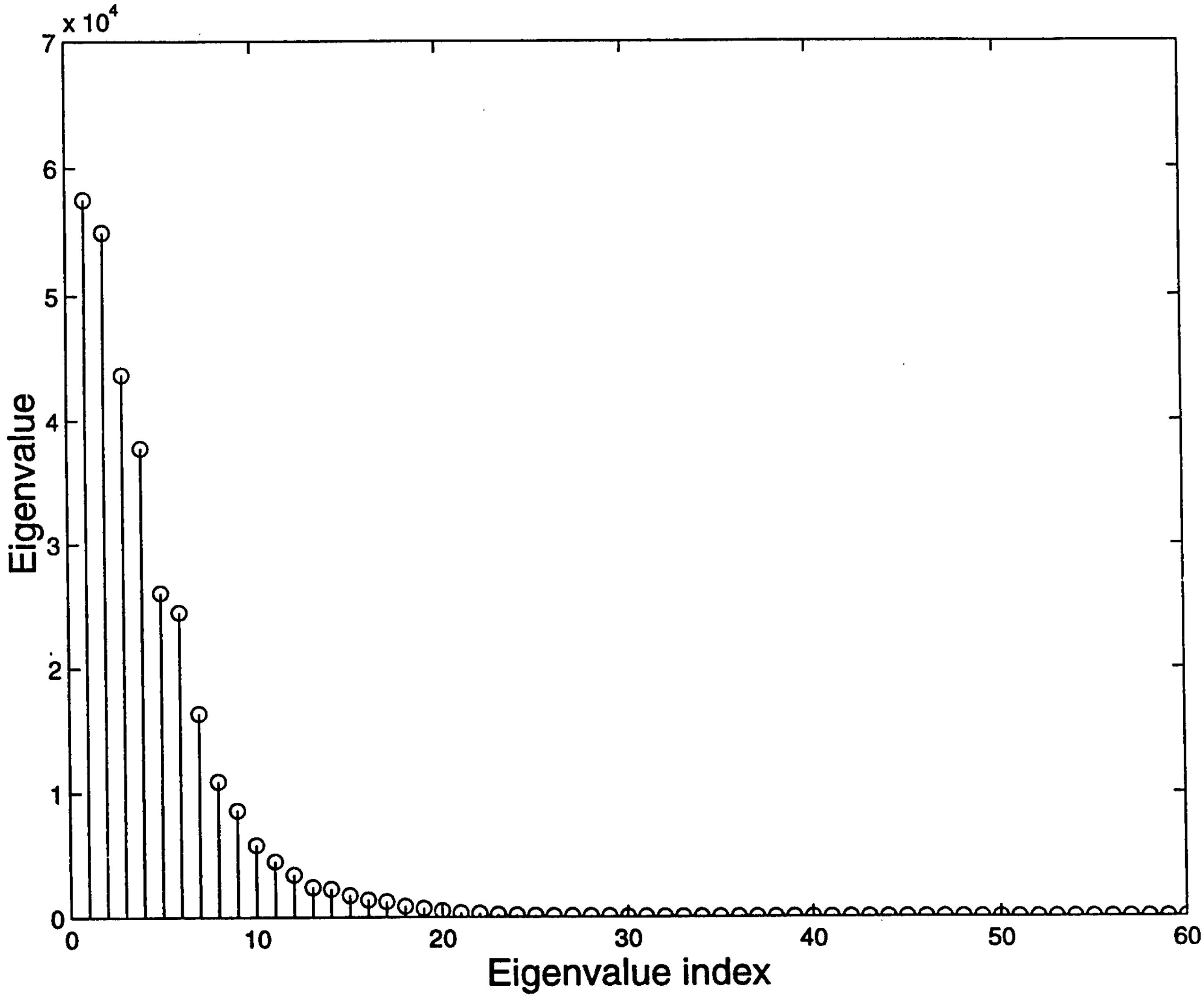


Figure 8: Eigen profile of clutter covariance matrix in Fig. 4.

Information Warfare: Faster Damage Assessment Through Efficient Logging

Brajendra Panda
Assistant Professor
Department of Computer Science

University of North Dakota
P.O. Box 9015
Grand Forks, ND 58202-9015

Final Report
for
Summer Research Extension Program
Rome Research Site

Sponsored by:

Air Force Office of Scientific Research
Bolling Air Force Base
Washington, D.C.

and

Rome Research Site
Air Force Research Laboratory
Rome, NY 13440

March 1999

Information Warfare: Faster Damage Assessment Through Efficient Logging

Brajendra Panda
Assistant Professor
Department of Computer Science
University of North Dakota

Abstract

System invasion is a common phenomenon in today's world specially after the concept of information sharing paved its way into the technical and business world. This certainly affects databases and brings them to an unstable and inconsistent state. It is required to recover and make the database operable as soon as such an attack is detected. However, extensive use of log is required during the damage assessment and recovery process. Log access time becomes a significant cost in this process due to the massive amount of data stored in the log, and the page I/O costs involved with it. In this research, we have developed algorithms to cluster the log based on transaction read-from relationships. This helps in eliminating the requirement of accessing the entire log, thus considerably reducing the damage assessment and recovery time. We prove that only one of these clusters will be accessed during damage assessment and recovery process. Through simulation we have confirmed that our model outperforms the traditional logging method in assessing the damage.

Information Warfare: Faster Damage Assessment Through Efficient Logging

Brajendra Panda

1. Introduction

Sharing brings growth. This truth brought into picture the concept of resources and information sharing. To sustain in the incredibly changing technological world, we need to share information with others. Computers are probably the most powerful information sharing devices in today's world and thanks to the Internet to make the process of information sharing even faster and easy. But every virtue brings a vice along with it. When we use our system to share information with others via network, we leave the door open for hackers to invade our system. It is extremely difficult to find and close all security flaws in the system. Hackers are always in search of new ways to prevail over the system security. Password sniffing and session hijacking are among various other means of intruding into a system. In such cases, the system under attack will not be able to detect an attacker from a legitimate user.

Defensive information warfare, which prepares the system to withstand such attacks and provide system integrity, is based on the model of prevention, detection, and recovery. A lot of research has been performed including strict access control mechanisms to protect the system from unauthorized users. Since prevention issues are outside the scope of this research, we would not reference them here. There are various intrusion detection techniques available to observe suspicious activities of users, and also detect any unauthorized access. Among them, a knowledge-based approach is presented in [7], while a statistical approach is offered in [8]. McDermott and Goldschalg have discussed a method called storage jamming ([10], [11]), that uses fake data termed as "honey pots" to attract attackers and flush them out. Several other techniques have been discussed in [9]. However, not all these methods detect an attack as it occurs. In most cases, the attacker enters the system and accesses data as a valid user. Thus, any update performed by the attacking transaction is made permanent when the transaction commits. Therefore, the damage can spread to other parts of the database through legitimate users as they update fresh data after reading any damaged data ([1], [5]). Consequently, timely recovery of the database is crucial to stop the cascading effect of the damage. Since operations of each transaction is stored in system log, and recovery algorithms that are designed for defensive information warfare scenarios require that the log must never be purged, the size of the log becomes massive. Therefore, the log access time during damage assessment and recovery becomes a significant factor.

In this research, we present algorithms to divide the log into several clusters based on transaction dependencies. We prove that only one of these clusters will be accessed during damage assessment and recovery process. We have used a simulation model to test the performance of our method. The result shows dramatic improvement over the time required to access traditional logs.

The rest of the report is organized as follows. In section 2, we discuss the motivation for this research. The clustering algorithm and its proof of correctness are presented in section 3. Section 4 offers an algorithm for the post clustering logging of transactions. The simulation model and performance comparisons with traditional logging method are offered in section 5. Section 6 concludes the report.

2. Motivation

Traditional recovery mechanisms as discussed in [2], [3], [4], and [6] have their scope confined to system or hardware failures. They are designed to flush out all effects of all non-committed transactions while ensuring that effects of all committed transactions are saved in the stable storage. For this kind of recovery, the vitality of read operations does not exist. Recovery is possible only from the before and after images of data items. However, as stated in [12], after the detection of an attack, effects of all transactions reading directly or indirectly from the malicious transaction have to be undone along with that of the malicious transaction. Then all affected transactions must be re-executed to reflect the consistent state of the database. Hence, in order to establish read-from relationships all read operations of transactions need to be stored in the log and the log can not be purged. A recovery protocol based on data dependency approach has been presented in [13] and [14], which allows affected transactions to be partially re-executed during recovery, therefore, incurring a faster recovery process. In any case, the system log must be accessed and the log must be read sequentially starting from the appearance of the attacking transaction in the log. This process requires unnecessary read of all unaffected transactions that appear after the malicious transaction, thus incurring too many superfluous page I/Os. Since page I/Os play the most dominating factor in data processing, avoidance of these activities as much as possible is essential.

3. Log Clustering

In this section, we describe a method for dividing the log into several clusters. This will help us identify the cluster to scan during damage assessment and recovery as opposed to the entire log. The principal idea of our model is that the part of the log that contains the attacking transaction need to be pinpointed, and only the clusters that contain affected transactions need to be accessed. This will save considerable amount of time during both damage assessment and recovery. Our model is based on the assumption that the history produced by transaction executions is strict. Next, we define some terms, which are used in our model.

Definition 1: A transaction T_j is *dependent upon* transaction T_i in a history if there exists a data item x such that: (1) T_j reads x after T_i updates x, and (2) every transaction that updates x in between T_i updates x and T_j reads x should be aborted before T_j reads it. It is needless to mention that, since the history is strict, T_i commits before T_j reads x.

Definition 2: If a transaction T_j is dependent upon transaction T_i, then we denote transaction T_j as the *dependent transaction* and T_i as the *precursor transaction.*

Transaction dependencies can be established by maintaining auxiliary data structures. In this research, we do not discuss any specific ways of doing so; rather, we focus on the recovery model. Our first algorithm takes the existing log as input and uses transaction dependencies to divide the log into various clusters. However, it is possible to establish dependency relationships while the transactions are executing. With this second approach, it would not be known whether a particular transaction will commit or abort. If a transaction aborts and it is already inserted to a cluster, then the transaction must be removed from the cluster, thus wasting a lot of effort. An important point to mention here is that if T_j is dependent on transaction T_i , then transaction T_i can't be dependent on transaction T_j since it would then violate the serializability of the history.

For the clustering algorithm to keep the dependent transactions in one cluster, we use a temp_set to store the transactions until a dependency is found. We also use another data structure, the cluster_list, where each cluster id and all transaction ids belonging to the corresponding cluster are kept. The copy operation used in the algorithm must preserve the order of operations as they appear in the log. This will preserve the original dependency by maintaining the partial order among transaction operations.

1. Set the temp_set = {}; //Empty set
2. Set Cluster_list = {}; Set Last_write = {};
3. Read all committed transactions from the log
   - 3.1 Check each operation o_i of every transaction T_i for dependency
     - 3.1.1 If no dependency found
       - 3.1.1.1 If T_i is not already in a cluster

         Write operation o_i to the temp_set;
       - 3.1.1.2 Else

         Write operation o_i to the cluster T_i is in;
     - 3.1.2 Else
       - 3.1.2.1 If both T_i and its precursor transaction, say T_j, are in temp_set

         Create a cluster;

         Copy all operations of T_j and T_i from the temp_set to the cluster;

         Copy operation o_i to the cluster;
       - 3.1.2.2 Else if only the precursor transaction T_j is in a cluster

         Copy all operations of T_i from temp_set to the same cluster;

         Copy operation o_i to the cluster;
       - 3.1.2.3 Else if T_i is in a cluster and the precursor transaction T_j is in the temp_set

Copy all operations of the T_j from the temp_set to the same cluster T_i is in;

Copy o_i to the cluster;

3.1.2.4 Else

3.1.2.4.1 If both T_i and T_j are in the same cluster

Copy o_i to the cluster;

3.1.2.4.2 Else

Copy all operations of all transactions from the cluster of T_j to the cluster of T_i;

Copy o_i to its cluster;

Remove the cluster T_j was in, and also remove the cluster entry from the cluster_list;

3.1.3 If o_i is a write operation ($w_i[x]$) for data item x

3.1.3.1 If x is already in the last_write table

Overwrite the entry with the new value x, T_i and the cluster number of T_i;

3.1.3.2 Else

Insert $w_i\,[x]$ and the cluster number of T_i as a new entry in the last_write table

Comments: In step 3.1.1.1, we check if the transaction already belongs to a cluster. This is possible when a prior operation of T_i would have established a dependency thus inserting T_i into a cluster. Both T_i and T_j may be in temp_set, as checked in step 3.1.2.1, when T_j is independent of any transaction, and none of the prior operations of T_i have established a dependency for T_i. Step 3.1.2.3 checks if the precursor transaction T_j is in the temp_set, but T_i is in a cluster. This could occur when T_j is independent of any transaction, where as for T_i, another pre-cursor transaction was found earlier. Due to the same reason, i.e., when T_i depends on multiple transactions, both T_i and T_j may be found in two different clusters. This is checked in step 3.1.2.4.2. Also note that, in step 3.1.2.4.2, when all operations of all transactions in one cluster are copied into another cluster, the partial dependencies among the transactions in both clusters are preserved.

Proof of Correctness

***Lemma 1*:** All operations of all committed transactions are checked to determine the dependency.

Proof: It is obvious from the step 3.1 of the clustering algorithm, where the algorithm checks all operations in the log. Step 3 makes sure that only the committed transactions from the log are taken into consideration for clustering algorithm.

***Lemma 2*:** Any transaction that has a dependency relationship with another transaction ends up in a cluster.

Proof: As the log is assumed to keep all the operations of a transaction, we check operation by operation of all the committed transaction to establish the dependency. If O_i is an operation, then in 3.1.1.1 the algorithm checks whether that operation establishes any dependency. If it is not and it is not in a cluster already it stays in the temp_set according to 3.1.1.1 until T_i establishes any kind of dependency or some other transaction establishes dependency with T_i making it a precursor transaction. If the transaction T_i is already in a cluster then the operation O_i is added to the cluster. 3.1.2.1 deals with the case if dependency takes place between T_i and T_j and they are in temp_set, a new cluster is built and these two transactions are kept there. In subsequent section 3.1.2.2 and 3.1.2.3 the algorithm deals with cases where only precursor transaction is in a cluster and only dependent transaction is in a cluster respectively. In 3.1.2.4.2 the algorithm deals with a special case where a transaction T_j, which is already in a cluster establishes dependency with another transaction T_k in another cluster. In that case it is suggested to merge the two clusters into one. Therefore, any precursor or dependent transaction will end up in a cluster.

***Lemma 3*:** If a transaction is neither dependent nor precursor of any transaction will stay in the temp_set.
Proof: Step 3.1.1.1 checks and stores each operation of each transaction in the temp_set if it is not establishing any dependency and the transactions have not become a *precursor* or *dependent* before. The entire block 3.1.2 makes sure that any transaction, which is a part of any dependencies, will end up in a cluster. The above argument suggests that at the end of the execution of the clustering algorithm, any transaction that is neither a precursor nor *dependent* transaction will reside in the temp _set.

***Lemma 4*:** A transaction can belong to only one cluster.
Proof: Imagine a transaction T_j is already in a cluster with its precursor transaction T_i. Then an operation O_j of transaction T_j establishes a dependency with transaction T_k, which is in another cluster. Section 3.1.2.4.2 of the clustering algorithm suggests copying all the operation of all the transactions from the cluster of T_k to the cluster of T_j and T_i. Then we remove the cluster of T_k and also remove the cluster from the cluster list. So the idea is to merge the clusters if a transaction in one cluster belongs to multiple clusters. This proves our claim that a transaction will belong to only one cluster.

For recovery we are only concerned about the read-from relationships [3] of transactions, so it is important that our cluster maintains all the read-from relationships as in the original log. The following theorem establishes this.

***Theorem 1*:** The read-from relationships among transactions remain the same as in the original log.
Proof: Assume that transaction T_j has a read from relationship with transaction T_i ($T_i \rightarrow T_j$) in the original log, T_i has written a data item, say x, which is read by T_j without any committed transaction updating x in between. If the cluster that contain T_i and T_j shows $T_k \rightarrow T_j$, then T_k has updated x after T_i updated x and before T_j read x. This means the copy of $w_k[x]$ from log to cluster is performed after the copy of $w_i[x]$ and before that of $r_j[x]$. Since the operations are copied in the same order as they appear in the log, the order of their appearance in the log is $w_i[x]$ <

$w_k[x] < r_j[x]$. This contradicts the fact that T_i has read from relationship with transaction T_j. This proves the theorem.

***Theorem 2*:** During damage assessment and recovery only one cluster will be accessed for one attacker.

Proof: According to *Lemma 4* a transaction will remain in one cluster only. If a transaction establishes a dependency, we merge the clusters of both precursor and dependent transactions. According to *Theorem 1* all the read-from relationships are kept intact, which means all transactions having read-from relationships with an attacking transaction will be in the same cluster as the attacking transaction. Therefore, during damage assessment only one cluster will be accessed for one attacker.

4. Post-clustering scenario

After the execution of the clustering algorithm, all the dependent and precursor transactions will be in some clusters. Clusters_list will have the updated cluster list for all the transactions. Last_write table will contain the last updated entries for every data item. Temp set will have all the transactions, which are neither *precursor* nor *dependent*. Taking all this change into consideration, we have to develop a model for the post-clustering scenario. Requirement of a system log is the major issue during the post-clustering operations of transaction in the database as there will be no system log in the post-clustering scenario. Functionality of the system log is of primary importance for all traditional operations in a database and for all the traditional recovery techniques. In our model, we have suggested to using the temp_set as the active log and temp_set along with all the clusters will function akin to the system log for everyday affair of the database. Both active and committed transactions are going to reside in the temp_set as long as they have not established any dependency. Unlike during clustering, we are going to deal with active transactions here in the post-clustering scenario. So the clustering algorithm is not quite compatible for this changed environment. We developed another algorithm to deal with this situation, which is presented next.

Post Clustering Algorithm

This new algorithm uses the old data structures temp_set and last_write table used in the clustering algorithm. Along with this, we have introduced a new data structure called precursor_list. For each active transaction we suggest to maintain a precursor list in the transaction buffer whenever any dependency is established. Precursor list is kept in the buffer until the transaction aborts or commits and the transactions are stored in a cluster if the transaction has a precursor, otherwise the transaction remains in the temp_set. If transaction commits then only its operations are available to other transactions. As the history is strict, no transaction can read or write a data item accessed by an active transaction. The following algorithm is used to place transactions in appropriate clusters during commit time ([2], [3]) of the transactions.

1. For each operation o_i
   1.1 If o_i is a write operation $w_i[x]$

Add the operation to the temp_set;

1.2 If o_i is a read operation $r_i[x]$

Add the transaction id of the precursor of T_i to the precursor list of T_i;

Leave it in temp_set;

1.3 If o_i is an abort operation

Remove all the operation of transaction T_i from the temp set;

Delete the precursor list of T_I;

1.4 If o_i is a commit operation

Update the last_write table for all the write operations done by T_i on any data item.

1.4.1 If the precursor list of T_i is not empty

1.4.1.1 If there is one precursor transaction T_j of T_i in the precursor list and T_j in a cluster

Move all the operation of T_i to the cluster of T_j;

1.4.1.2 Else

1.4.1.2.1 If all the precursors are in temp_set

Create a new cluster;

Move all operations of the precursor transactions in temp_set to the cluster;

Move all operations of T_i to the cluster;

1.4.1.2.1 Else

Merge the clusters (if there are more than one) of the precursors to one cluster;

Move all operations of the precursor transactions in the temp_set (if any) to the same cluster;

Move all the operation of T_i from the temp_set to the same cluster;

2. Delete the precursor list of T_i.

Correctness of this algorithm is similar to that of the clustering algorithm and, therefore, omitted here.

5. Performance Analysis

We developed a simulation model and collected several sets of data to compare the number of transactions to be scanned and the time required to read these transactions from the secondary storage. The results obtained are highly promising. Our model clearly outperforms the traditional methods in all cases. Next, we present a brief description of the simulation model.

In our simulation, transactions access data items randomly. The operation (whether read, or write, or read/write) is also randomly chosen. The simulation program keeps a read and a write set for each transaction. It maintains a data structure called last_write for each data item to store the most recent write operation performed on them. When there is a read operation on any data item, the last_write table is checked to see which transaction wrote the item last. This establishes the dependency between the transaction (precursor) which wrote the data item last and the transaction (dependent) reading the data item. Once the dependency is established, the ids of the precursor and the dependent transaction are written to a cluster. Cluster information is maintained by a data structure called cluster_list.

We ran our simulation on two varying parameters. First, the maximum number of data items accessed by the transactions was changed and the total number of available data items was kept fixed. Then we kept the maximum number of data items accessed by each transaction and ran our simulation with changing total number of available data items. We ran these simulations for two different attackers (T_{10} and T_{20}) for the two changing parameters discussed above. Then for each attacker, we plotted two graphs comparing the number of transactions to be scanned for recovery and the total page I/O time to demonstrate the performance of our model and the log based model. Table 1 lists the system dependent fixed parameters.

Variable	Description	Value
SC_i	Space taken by the commit operation of a transaction in the log	12 bytes
SS_i	Space taken by the start operation of a transaction in the log	12 bytes
SR_i	Space taken by a read operation of a transaction in the log	40 bytes
SW_i	Space taken by a write operation of a transaction in the log	60 bytes
PS	Page Size	2 KB
PT	Page I/O Time	15 milliseconds

Table 1: System Dependent Parameters

For the log based recovery method, the recovery process begins by scanning all operations of transactions from the point of attack to the end of the log. So to calculate the total page I/O for this model we calculated all the read, write, start and commit operations of transactions starting from the attacking transaction to the last transaction in the log. By using the space occupied by each of these operations as listed in Table 1, we calculated the size of the part of the log that needs to be scanned and hence the total page I/O time. The following example explains our calculation.

Example: Assume that the total number of transactions in the log is 30 and the attacker is the 10^{th} transaction. So we need to scan 21 transactions in the traditional method. If these 21 transactions have a total of 150 read operations and 200 write operations, we compute the total page I/O time for the traditional method as follows.

Total space for read operations: 150 * SR_i = 6000 bytes

Total space for write operations: 200 * SW_i = 12000 bytes

Total space for commit operations: 21 * SC_i = 252 bytes

Total space start operations: 21 * SC_i = 252 bytes

Total Space needed to be scanned: 6000 + 12000 + 252 + 252 = 18504 bytes

Number of pages needed: $\lceil$total space / PS$\rceil$ = 10

Total page I/O time: 10 * PT = 150 milliseconds

Above mentioned steps have been used to calculate the total page I/O or Log Access Time for the traditional log based methods. To calculate the total page I/O for our method, we used the same steps but the number of transactions to be scanned is determined from the appropriate cluster.

In figures 2 through 5, the total number of transactions and the total number of data items are kept constant whereas the maximum number of data items, which can be accessed by each transaction, has been varied. In figures 1 and 2 the 10^{th} transaction is assumed to be the attacker. In figures 3 and 4 the attacker is chosen to be the 7^{th} transaction.

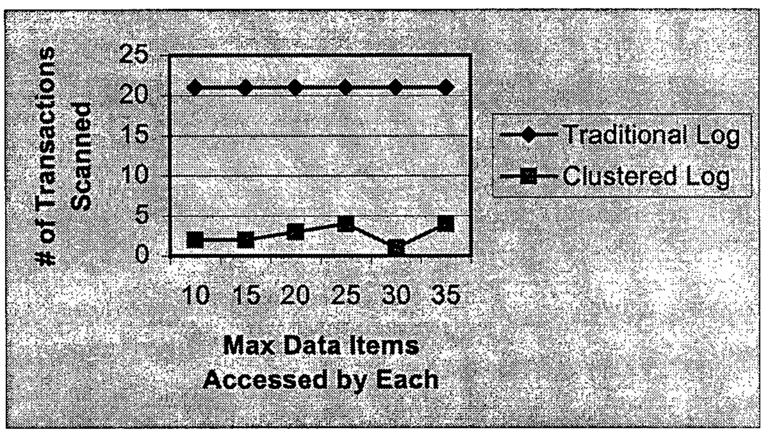


Figure 2: Total Transactions to be Scanned When Attacker is T_{10}

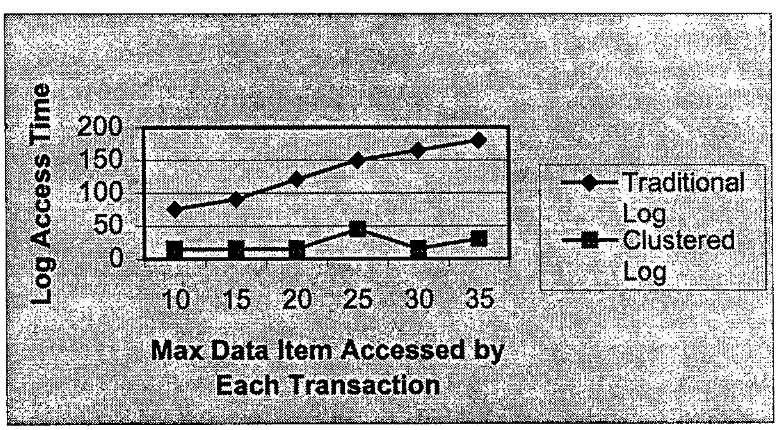


Figure 3: Log Access Time Comparison When Attacker is T_{10}

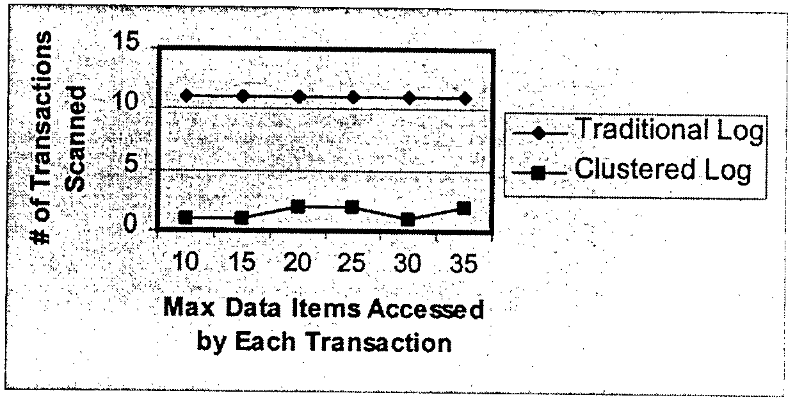


Figure 4: Total Transactions to be Scanned When Attacker is T_{20}

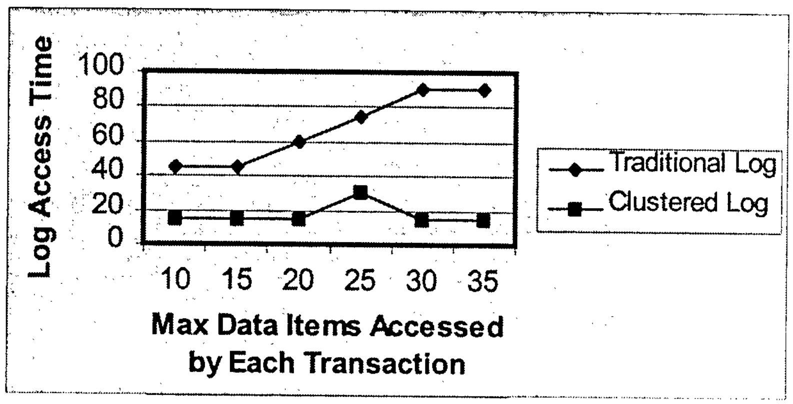


Figure 5: Log Access Time Comparison When Attacker is T_{20}

In figures 6 through 9, we have kept total number of transaction and max number of data items accessed fixed. We plotted the graphs with varying total number of data items available. The attacker is chosen to be the 10th transaction in figures 6 and 7, and as the 20th transaction in figures 8 and 9.

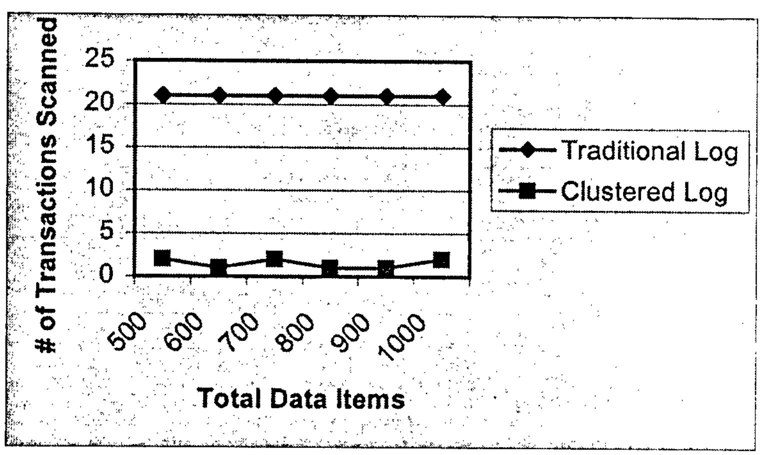


Figure 6: Total Transactions to be Scanned When Attacker is T_{10}

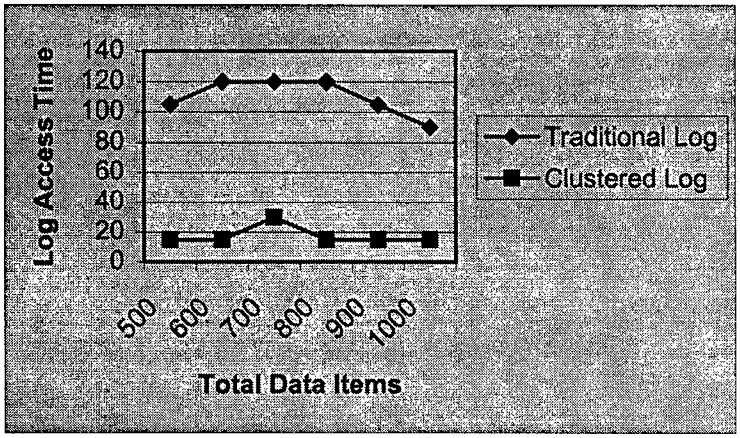


Figure 7: Log Access Time Comparison When Attacker is T_{10}

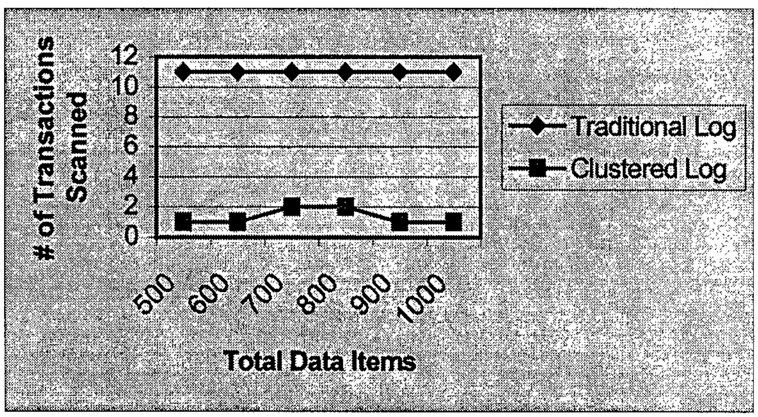


Figure 8: Total Transaction to be Scanned When Attacker is T_{20}

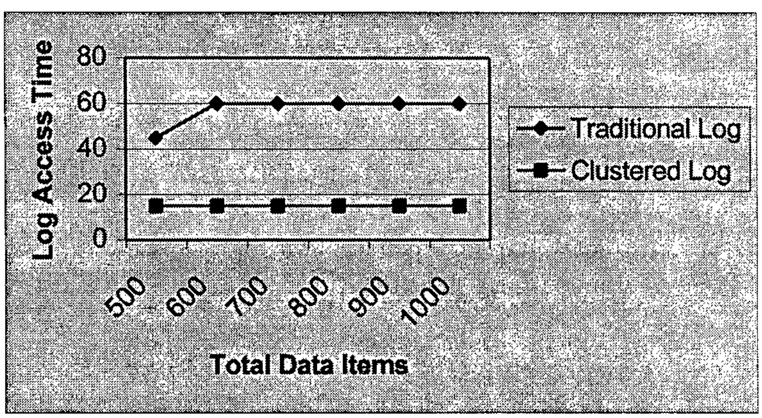


Figure 9: Log Access Time Comparison When Attacker is T_{20}

6. Conclusion

Every computer system that is networked is vulnerable to information attacks. While retaliation to these attacks is virtually impossible, immediate attention must be paid for database survivability. Quick and efficient recovery of the system is vital for an organization whose information resources have been attacked. In this research, we have presented two algorithms that cluster the log based on transaction dependency. The first algorithm divides an already existing log into several clusters. This is necessary to remodel conventional database management systems to adapt to log clustering method. The second algorithm uses already clustered log to place committed transactions in appropriate cluster. We expect a background process to perform this activity for improved efficiency. We have used simulation to verify the performance of our method. Several graphs are presented to depict the results, which show significant improvement in damage assessment and recovery time using our model.

Acknowledgments

I acknowledge and thank Mr. Joseph Giordano of Rome Research Site, Air Force Research Laboratory for his support of during this research. I am also thankful to Mr. Satyadeep Patnaik for helping me perform this research.

References

[1] P. Ammann, S. Jajodia, C. D. McCollum, and B. Blaustein, "Surviving Information Warfare Attacks on Databases", In Proceedings of the 1997 IEEE Symposium on Security and Privacy, p.164-174, Oakland, CA, May 1997.

[2] P. A. Bernstein, V. Hadzilacos, and N. Goodman, "Concurrency Control and Recovery in Database Systems", Addison-Wesley, Reading, MA, 1987.

[3] R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems", Second Edition, Addison-Wesley, Menlo Park, CA, 1994.

[4] J. Gray and A. Reuter, "Transaction Processing: Concepts and Techniques", Morgan Kaufmann, San Mateo, CA, 1993.

[5] R. Graubart, L. Schlipper, and C. McCollum, "Defending Database Management Systems Against Information Warfare Attacks", Technical report, The MITRE Corporation, 1996.

[6] H. F. Korth and A. Silberschatz, "Database System Concepts", Second Edition, McGraw-Hill, New York, 1991.

[7] T. F. Lunt, R. Jagannathan, R. Lee, A. Whitehurst, and S. Listgarten, "Knowledge-based Intrusion Detection", In Proceedings of the 1989 AI Systems in Government Conference, March 1989.

[8] T. F. Lunt, "Using Statistics to Track Intruders", In Proceedings of the Joint Statistical Meetings of the American Statistical Association, August 1990.

[9] T. F. Lunt, "A Survey of Intrusion Detection Techniques", Computers & Security, Vol. 12, No. 4, p. 405-418, June 1993.

[10] J. McDermott and D. Goldschlag, “Storage Jamming”, Database Security IX: Status and Prospects, D. Spooner, S. Demurjian, and J. Dobson, editors, p. 365-381, Chapman & Hall, London, 1996.

[11] J. McDermott and D. Goldschlag, “Towards a Model of Storage Jamming”, In Proceedings of the IEEE Computer Security Foundations Workshop, p. 176-185, Kenmare, Ireland, June 1996.

[12] P. Ammann, S. Jajodia and P. Liu, "Recovery From Malicious Transactions", Technical report, Dept. of Information and Software Engineering, George Mason University, 1997.

[13] Brajendra Panda and Joeseph Giordano, “An Overview of Post Information Warfare Data Recovery”, In Proceedings of the 1998 ACM Symposium on Applied Computing, Atlanta, GA, February 1998.

[14] Brajendra Panda and Joseph Giordano, “Reconstructing the Database after Electronic Attacks”, In Proceedings of the IFIP 11.3 12th Annual Conference on Database Security, Porto Carras Complex, Chalkidiki, Greece, July 1998.

COMPLEXITY OF DETECTING AND CONTEXT-DRIVEN METHODS FOR RESOLVING DATABASE INCONSISTENCY

Michael A. Pittarelli
Associate Professor
Department of Computer Science

State University of New York
Institute of Technology at Utica/Rome
P.O. Box 3050
Utica, NY 13504-3050
mike@cs.sunyit.edu

Final Report for:
Summer Faculty Research Program
Air Force Research Laboratory
Rome Research Site

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, Washington, D.C.

December, 1998

COMPLEXITY OF DETECTING AND CONTEXT-DRIVEN METHODS FOR RESOLVING DATABASE INCONSISTENCY

Michael A. Pittarelli
Associate Professor
Department of Computer Science
SUNY Institute of Technology at Utica/Rome

Abstract

Detection of inconsistency of frequency databases (collections of contingency tables) is an NP-complete problem. As has been demonstrated for other NP-complete problems, it is possible to identify the hardest instances of the consistency detection problem in terms of an order parameter analogous to those for phase transitions in physical systems. In addition to developing an order parameter, this report presents methods for utilizing inconsistent databases in decision making.

COMPLEXITY OF DETECTING AND CONTEXT-DRIVEN METHODS FOR RESOLVING DATABASE INCONSISTENCY

Michael A. Pittarelli

I. Introduction

Much work has been done in recent years on the problem of identifying hard instances of constraint satisfaction problems [1]. Most of these problems are NP-complete [2]. Although such problems (probably) require exponential time in the worst case, in practice, many instances can be solved reasonably quickly.

The research on locating hard instances of these problems has focussed on identifying critical parameters in the description of an instance. For example, for randomly generated instances of 3-SAT, the difficult cases are concentrated around a particular ratio of clauses to variables [3]. For the problem of determining whether a partially specified latin square (operation table for a quasigroup) can be completed, there is a value for the percentage of entries that are fixed in advance around which the difficult cases are clustered [4]. There tends to be an abrupt change in the performance of algorithms as one goes from instances characterized by values less than the critical value to instances with greater values. This behavior is analogous to a phase transition in a physical system.

The type of problem dealt with in this report is that of determining whether a collection of partially specified tables of frequency distributions (a *frequency database*) is consistent. The tables below represent a (completely specified) frequency database that is consistent:

Gender	Dog_Owner	$f(.)$
Male	Yes	20
Male	No	10
Female	Yes	30
Female	No	40

Dog_Owner	SUV_Owner	f'
Yes	Yes	12
Yes	No	38
No	Yes	4
No	No	46

The database is consistent because the two tables agree on the frequencies for the shared attribute "Dog_Owner"; it can be inferred from either table that there are 50 dog owners in the population corresponding to the database.

A completely specified database is a special case of a partially specified database. For a partially specified database, 0 or more of the frequencies are missing. For example:

Gender	Dog_Owner	$f(.)$
Male	Yes	?
Male	No	10
Female	Yes	?
Female	No	?

Dog_Owner	SUV_Owner	f'
Yes	Yes	12
Yes	No	?
No	Yes	4
No	No	46

The database is consistent, since it can be completed in such a way that there exists a frequency distribution over the three attributes Gender, Dog_Owner and SUV_Owner whose marginals over the pairs of attributes {Gender, Dog_Owner}, {Dog_Owner, SUV_Owner} coincide with those given in the partially specified tables. (The concept of consistency is more thoroughly discussed in Section III, which also contains formal definitions of various other database concepts.)

Section II of this report details the work done on attempting to develop a measure (order parameter) that would allow prediction, based on parameters describing the problem instance, of whether a partially specified database instance is likely to be consistent and also of the relative difficulty of determining whether the instance is consistent.

An equally important component of the research conducted deals with the use to which a database of marginal frequencies could be put. Statistical frequencies are normally used as the basis for estimating probabilities. Probabilities, in turn, are typically used as the basis for decision making. In Section III, methods for decision making using possibly inconsistent frequency tables will be discussed.

II. Experiments

We conducted experiments to locate difficult instances of the problem of determining whether partially specified frequency database tables are consistent. The experiments involved binary attributes and database schemes of a very restricted form, as described below. The general framework, for the case of 3 attributes, is as follows:

A_1	A_2	f_1
0	0	x_8
0	1	x_9
1	0	x_{10}
1	1	x_{11}

A_2	A_3	f_2
0	0	x_{12}
0	1	x_{13}
1	0	x_{14}
1	1	x_{15}

A_1	A_2	A_3	f_0
0	0	0	x_0
0	0	1	x_1
0	1	0	x_2
0	1	1	x_3
1	0	0	x_4
1	0	1	x_5
1	1	0	x_6
1	1	1	x_7

The f_i are frequency distributions. The constraints are:

$$x_i \geq 0$$
$$x_0 + x_1 + \cdots + x_6 + x_7 = N$$
$$x_8 = x_0 + x_1$$
$$x_9 = x_2 + x_3$$
$$\cdots$$
$$x_{14} = x_2 + x_6$$
$$x_{15} = x_3 + x_7$$

Note that these equations imply the constraints: $\sum_{i=8}^{11} x_i = N$, $\sum_{i=12}^{15} x_i = N$.

Zero or more of the marginal probabilities $x_8, \ldots, x_{15}$ have values preassigned to them. (The case in which all of the marginals have values assigned to them was proved in 1977 to be NP-complete [5].) None of the joint probabilities have values preassigned. The problem is to determine whether values satisfying the constraints can be found for the remaining variables. If so, then the partially specified marginal frequency tables are *consistent*. Consistency can also be defined for partially specified tables in the absence of a specified sum N. In that case, the tables are consistent if there exists at least one fully specified joint table (involving each of the attributes in the marginal tables) whose marginals agree with those given in the marginal tables. (Inconsistency will be discussed more thoroughly in Section XXXX. Section XXXXXXXXX discusses methods for utilizing inconsistent frequency tables.)

The number of variables can be reduced. The marginal tables can be reexpressed as:

A_1	A_2	f_1	A_2	A_3	f_2
0	0	$x_0 + x_1$	0	0	$x_0 + x_4$
0	1	$x_2 + x_3$	0	1	$x_1 + x_5$
1	0	$x_4 + x_5$	1	0	$x_2 + x_6$
1	1	$x_6 + x_7$	1	1	$x_3 + x_7$

Notice that in each subdistribution, each of the x_i appears exactly once. (This is not the case for more general database schemes.) Also, for database schemes like this one, of the form

$$\{\{A_1, A_2, \cdots, A_{n-1}\}, \{A_2, A_3, \cdots, A_n\}\},$$

each sum

$$x_i + x_j$$

is of one of two simple forms:

a) $i \in \{0, 1, 2, \cdots, 2^{n-1}-1\}$ and $j = i + 2^{n-1}$

b) $i \in \{0, 2, 4, \cdots, 2^n-2\}$ and $j = i + 1$.

The maximum number of marginal constraints is always 2^n, which equals the number of joint tuples (i.e., the number of variables).

In the latin square completion problem, the parameters are the number of rows and columns and the number of preassigned entries. The number of different values for the entries is equal to the number of rows. For the 3-SAT problem, the parameters that can be controlled are the number of propositional variables and the number of clauses. For this problem, there are three parameters (vs. two): the number of attributes, the number of preassigned marginal frequencies, and the value of N.

In early experiments, it was attempted to control for the interaction of N with the other two parameters by keeping the sum of the number of partitions of N into 2^n parts constant. For the numbers of attributes in the experiments conducted ($n = 4,5,6,7$), $N = 32$ kept this quantity approximately constant. (Note: A more "realistic" value might have been something like 5×2^n - the recommended minimum number of observations for the applicability of certain statistical procedures. However, the set of possible values $\{0,1,...,N\}$ for the variables would have been too large to conduct the experiments. For SAT problems, the variables have only two possible values, vs. N+1. For the latin square problem, the number of values is equal to the number of rows, but results have been reported for squares with at most 20 rows.)

The difficult instances of latin square completion problems occur when approximately 43% of the elements are preassigned, regardless of the number of rows. Similarly, for 3-SAT problems, the hard cases arise when the ratio of clauses to variables is approximately 4.3, regardless of the number of variables. These problems lack an analogue of the global constraint that the sum of all of the variables equals N.

This combination of a relatively weak constraint involving all of the variables and constraints involving pairs of variables (values for specified marginal frequencies, equal to the sum of a pair of joint frequencies) complicated the search for a phase transition, i.e., a way of identifying, independently of the number of attributes, the point at which determining whether or not a pair of frequency tables is consistent becomes especially difficult.

This and other attempts at controlling for factors other than the percentage of preassigned marginals (thus allowing its use as an order parameter) were unsuccessful.

Another fundamental problem involved the fact that the number of joint frequencies is an exponential function of the number of attributes for a given problem instance (2^n, where n is the number of attributes). This, combined with the desire for a realistic average marginal value, severely limits the number of problem ensembles that can be generated.

A solution (suggested by Bart Selman) to the problem of explosive growth of the state space as the number of attributes is increased is to keep the number of attributes fixed at 3, but vary the domain size of one of them (a "superattribute"). When the domain size of the superattribute is an integer power (i) of 2, the problem is isomorphic to a multiattribute problem with i attributes. For the original problem instances, the possible numbers of joint frequencies are:

$$8, 16, 32, 64, 128, 256, 512, \ldots$$

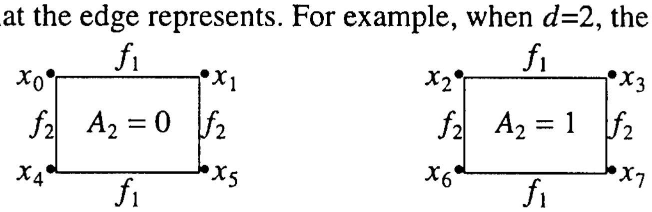
For the "superattribute" version, the possible numbers of joint frequences are multiples of 4:

$$8, 12, 16, 20, 24, 28, 32, \ldots$$

With database scheme

$$\{\{A_1, A_2\}, \{A_2, A_3\}\},$$

where A_1 and A_3 are binary and A_2 is d-ary, the joint tuples are partitioned into d equivalence classes, where two joint tuples are in the same class if they agree on the value for (have the same projection onto) the superattribute (A_2). There is a total of $4d$ marginal frequencies. Each marginal frequency is the sum of (constrains) two joint frequencies. (Similarly, the marginal tuples are partitioned into d equivalence classes). Again, there are two marginal frequency tables: f_1 over A_1 and A_2, and f_2 over A_2 and A_3. The structure of the constraints may be represented as an undirected graph whose edges represent marginal frequencies and whose nodes represent joint frequencies. Two nodes are connected by an edge if their sum is the value of the marginal frequency that the edge represents. For example, when d=2, the graph is:



x_0 is the frequency of tuple $A_1A_2A_3$ = 000, x_1 is the frequency of $A_1A_2A_3$ = 001, ..., and x_7 is the frequency of $A_1A_2A_3$ = 111.

The total number of marginal frequencies is $4d$ (as is the total number of joint frequencies). In the experiments that were performed, the number of randomly generated marginal frequencies varied, and never reached more than about 30% of $4d$. Thus, the problems are analogous to latin square (quasigroup) completion problems.

Any of the d graphs for the equivalence classes ($A_2 = 0, A_2 = 1, \ldots, A_2 = d$) may have 0 to 4 edges. If there are 0 edges, the joint frequencies are constrained only by the global constraints (nonnegativity, sum = N). If there is one edge, then exactly two joint frequencies are constrained by one marginal frequency. If there are two edges for a single equivalence class, then there are two possibilities:

(i) The marginals are from the same marginal distribution, in which case two disjoint sets of two joint frequencies are constrained. For example:

$$x_0 + x_1 = m_1$$
$$x_4 + x_5 = m_2$$

(ii) The marginals are from different marginal distributions, in which case three joint frequencies are constrained. For example:

$$x_0 + x_1 = m_1$$
$$x_1 + x_5 = m_2$$

The difference in the degree to which the overall solution is constrained in the two cases depends on N and on the values of the marginals. Assume that

$$m_1 = m_2 = k.$$

For the subproblem in which the joint frequencies constrained by the marginals are considered in isolation from any other constraint (e.g., that all the joint frequencies must sum to N), the number of solutions for case (i) is $(k + 1)^2$ However, as a fraction of size of the state space for this subproblem, this is

$$(k + 1)^2/(N + 1)^4$$

For case (ii), there are only $k + 1$ solutions; the value of any one of the three joint frequencies determines the values of the other two. So, the fraction of the elements of the (sub)state space that are solutions, in case (ii), is:

$$(k + 1)/(N + 1)^3$$

These fractions are plotted for $N = 256$ and $k = N/2d$, the average value of a marginal if each marginal is assigned a value, in Figure 1.

With 3 or 4 marginals in one equivalence class, all 4 joint frequencies are constrained. Again assuming that each marginal equals some constant k, the fraction of solutions in the (sub)state space is $(c + 1)/(N + 1)^4$, in either case.

Experiments were conducted for domain sizes of 10, 12, 14, 16 and 18 for the superattribute. For each domain size, the number of marginals randomly assigned values was varied from about 2% of the total number of marginals to about 40%. (Although the percentage of marginals assigned is not an order parameter for this problem, as discussed below, for each domain size the transition from nearly all consistent to nearly all inconsistent occurred well before 40% of the marginals were assigned values.)

In Figure 2, the 25th percentile of the number of backtracks required to determine whether or not a database is consistent is plotted against the percentage of marginals selected and given values. The procedure used to determine consistency was a reverse Brelaz heuristic, implemented in C++, using ILOG Solver libraries [6, 7]. The program was adapted from programs developed by Carla Gomes for the quasigroup completion problem [4]. (Figures 3 and 4, respectively, plot backtracks for domain sizes 12 and 16 separately.) As Figure 2 illustrates, the percentage of marginals selected at which the number of backtracks is maximized decreases as the domain size increases. (In the experiments conducted, from 20.0% for domain size = 10 to 18.1% for domain size = 18.) Therefore, the percentage of marginals selected does not appear to be an order parameter for the database consistency problem.

Following the unsuccessful attempts to keep invariant the degree to which a problem ensemble is constrained, by keeping constant the number of solutions in the case in which none of the marginals are preassigned, we became aware of the work of researchers at the University of Leeds and Strathclyde University on characterizing the "constrainedness" of ensembles of combinatorial search problems. Gent et al. [8] measure the *constrainedness* of an ensemble of problems as

$$1 - \log_2(< Sol >)/\log_2(|S|)$$

where $< Sol >$ is the expected number of solutions for a given problem in the ensemble and S is the state space for the problem. Good results (and agreement with measures developed independently) are shown for this measure for a variety of combinatorial search problems: satisfiability, graph coloring, general constraint satisfaction, and number partitioning.

For the superattribute version of the database consistency problem, the size of the state space is

$$(N + 1)^{4d},$$

where N is the specified sum (of the joint frequencies and of the marginal frequencies, for both marginal frequency tables) and d is the domain size for the superattribute.

The expected number of solutions, $< Sol >$, can be calculated as the size of the state space times the fraction of solutions determined by the parameters for a problem ensemble (domain size, sum, number of preassigned marginals). For this problem, there is a global constraint (sum of joint frequencies = N) and p "local" constraints, one for each of the p preassigned marginals. The fraction of the state space satisfying the global constraint is

$$\frac{C(4d + N - 1, N)}{(N + 1)^{4d}}.$$

Considered independently, each of the p preassigned marginal values constrains two joint frequencies. Let m_j denote the (expected) value of the jth marginal. Then for the (sub)state space consisting of pairs of values for the affected joint frequencies, the fraction satisfying jth marginal constraint is

$$\frac{m_j + 1}{(N + 1)^2}.$$

On the assumption of independence of the marginal constraints, then,

$$< Sol > = (N+1)^{4d} \times \prod_{j=1}^{p} \frac{m_j + 1}{(N+1)^2} \times \frac{C(4d+N-1, N)}{(N+1)^{4d}}$$

$$= \prod_{j=1}^{p} \frac{m_j + 1}{(N+1)^2} \times C(4d+N-1, N).$$

Gent et al. refer to their constrainedness measure as kappa (κ), and we will refer to various refinements of this measure for this problem as kappa-0, kappa-1, etc. Kappa-0 will refer to the version in which: the marginal constraints are assumed independent of each other; the (expected) marginal values are assumed equal to each other; and the expected marginal value equals $N/2d$. Figure 5 plots backtracks against kappa-0 for domain sizes 10, 12, 14, 16, and 18. The "drifting to the left" of the peaks evident in Figure 2 does not occur in Figure 5. However, there is clearly room for improvement.

We will call "kappa-1" the refinement of kappa-0 in which the expected marginal is calculated taking into account the manner in which the marginals are generated experimentally. Kappa-0 assumes that the expected marginal value equals the table sum (N) divided by the number of marginal frequencies per table ($2d$). This assumption is correct if each marginal is preassigned. However, the percentage of preassigned marginals never exceeded 35.7% in the experiments conducted. Furthermore, in the experiments, each table was allocated N units to distribute (although not necessarily to exhaust) among the marginal frequencies selected from that table. For each marginal value randomly generated (with uniform probability over the integers from 0 to the number of allocated units remaining), the remaining units were reduced by that value. Thus, the expected value of the first marginal for a marginal table would be $N/2$; the expected value for the ith marginal from a table would be $N/2^i$. However, the program used for these experiments limited the marginal values to 8 times what the average marginal value would be if the marginal tables were completely specified: $4N/d$. Without this restriction on the range of marginal values, the problem ensembles generated tended not to show very dramatic transitions from easy to hard to easy regimes. Figure 6 plots backtracks against kappa-1, where values for kappa-1 are based on experimentally observed average marginal values.

The value of kappa-1 for a particular (N, d, p) triple is lower than that of kappa-0. The empirically observed average marginal value used in the calculation of kappa-1 is higher than the estimate used by kappa-0; thus the contribution of each marginal to $< Sol >$ is greater, in kappa-1: $m > m'$ implies

$$\frac{m+1}{(N+1)^2} > \frac{m'+1}{(N+1)^2}$$

Furthermore, the values of kappa-1 for the hardest cases for the two largest domain sizes (16 and 18) are equal (to 3 decimal places). However, the fact that the numbers of preplaced marginals are integers, and are somewhat small, reduces the significance of this result.

Kappa-1 still assumes that the marginals (for a given N, d and p) have the same expected values. Kappa-2 is calculated using separate expected values for each of the preplaced marginals. Although the variations are in some cases quite extreme (e.g., 6.8 for the 1st marginal, with N = 56 and d = 14, vs. 2.1 for the 20th marginal) the effect on the constrainedness measure is not very dramatic. The values for the hardest cases for the three largest domain sizes (14, 16 and 18) are in slightly better agreement. In general, the value of kappa-2 for any particular N, d and p is slightly smaller than the corresponding value for kappa-1. This is due to the fact that

$$\prod_{j=1}^{p}(m_j + 1) \leq (\sum_{j=1}^{n}(m_j + 1)/n)^n$$

Backtracks are plotted against kappa-2 in Figure 7.

The peak effort for domain sizes 10 and 12 occurs at lower values than for sizes 14, 16 and 18, for each version of kappa (-0, -1 and -2). For kappa-2, the peaks occur at 0.933 and 0.931 for d = 10 and d = 12, respectively, vs. 0.959, 0.957 and 0.956, respectively, for d= 14, d = 16 and d = 18.

A plausible explanation is that independence of the marginal constraints, which is assumed by each of these measures, is more likely for the smaller domain sizes, for the number of preassigned marginals generating the most backtracks. As discussed previously, for a problem instance with p preassigned marginals and domain size d, the p constraints are distributed among d equivalence classes. A sufficient condition for independence of the p marginals is that they are distributed among the d equivalence classes in such a way that no class is assigned more than one marginal. In other words, the mappings from marginals to equivalence classes is 1-1. The fraction of 1-1 mappings from marginals to classes,

$$\frac{d!/(d-p)!}{d^p},$$

is quite low (at the point of peak backtracks) for all the problem instances studied but does increase as the domain size for the superattribute decreases:

$$d = 18,\ p = 13\text{: } 0.003$$
$$d = 16,\ p = 12\text{: } 0.003$$
$$d = 14,\ p = 11\text{: } 0.004$$
$$d = 12,\ p = 9\text{: } 0.015$$
$$d = 10,\ p = 8\text{: } 0.018.$$

Furthermore, there is a steep increase in this value as d decreases from 14 to 12.

Therefore, the expected number of solutions is underestimated (and the degree of constraint is overestimated) to a greater extent for larger values of of d. For example, with two marginals in a particular class, the probability is 1/3 that the two constraints are independent. For d = 10, p = 8 and the observed average marginal value of 7.5, the solution fraction calculated by our measures, which assume independence, is 3×10^{-5}. However, if the two marginals are not independent (the probability of which is 2/3), the solution fraction calculated should be 1×10^{-4}. For d = 18 and p = 13, the fractions are 3×10^{-6} and $2 \times 10 - 5$. Although each difference is a single order of magnitude, the ratio is twice as large for d = 18.

Mappings from marginals to equivalence classes can be distinguished by the number of marginals occupying each class, where the number in each class is an integer in the range 0 to 4. Call an assignment of an integer in the range 0 to 4 to each equivalence class, where the sum of the integers is p, a configuration. Each configuration has a probability. With the simplifying assumption that each marginal (1st, 2nd, ..., pth, for a given d, p and N) has the same expected value, m, the expected number of solutions can be calculated for each configuration, weighted by the corresponding probability, and summed. For example, when $d = 2$ and $p = 3$, the equivalence classes are:

The number of configurations is 20: 4 in which class $d = 0$ is empty and class $d = 1$ has 3 members, in which case the marginals' contribution to the calculation of $< Sol >$ is

$$\frac{m+1}{(N+1)^4};$$

6 in which class $d = 0$ has one member and class $d = 1$ has 2 members, where the contribution to $< Sol >$ is

$$\frac{1}{3} \times \frac{(m+1)^2}{(N+1)^4} + \frac{2}{3} \times \frac{(m+1)}{(N+1)^3}; etc.$$

I have not implemented a version of constrainedness incorporating these refinements; I hope to work on this problem in the very near future with someone with expertise in physics (e.g., Bart Selman).

Gomes, Selman and Crato [9] have discovered that some instances of combinatorial search problems have extreme variability in the amount of computational effort required to solve them using different heuristics, or for stochastic methods, different random seeds. For some of these instances, the distributions are "heavy-tailed". Roughly, for a stochastic algorithm, the running mean of the measure of computing effort (time, number of backtracks) over repeated trials with the same instance does not converge. No instances of the database consistency problem were found with this behavior, despite spending hundreds of cpu hours in the search for such instances. The search was concentrated on instances near the peak number of backtracks for a given domain size. Furthermore, since consistent instances were much more likely to exhibit wide variation in the effort required to solve them, the search was eventually limited to such cases.

Eventually, in the course of searching for heavy-tailed problem instances, experiments were devised for database instances with α-cyclic schemes [10]. Such schemes are well-known to have undesirable properties, for example, with respect to efficient join processing. (Cyclicity of database schemes is discussed in Section III.D.)

For these experiments, there were n binary attributes and n marginal tables, each involving $n - 1$ of the attributes. Whereas the "constraint graph" for a "superattribute" database scheme with domain size d consists of d disconnected 4-node, 4-edge undirected graphs (see above), the constraint graph for these schemes may be visualized as 2^{n-3} cubes (12 marginals/edges each) plus edges connecting each vertex with 2^{n-4} vertices in other cubes (for $n \geq 4$; for $n = 3$, the graph is a single cube). So, for $n = 4$, the graph may be laid out so as to resemble a box kite; for $n = 5$, two loosely connected box kites; etc. Figure 8 plots median backtracks against the percentage of marginals preassigned, and figure 9 plots backtracks against kappa-1 for 6 attributes with $N = 96$, and 7 attributes with $N = 128$. (Note that the percentage at which the backtracks are maximum is halved when the number of attributes is doubled and that the percentages are much lower than for the superattribute experiments.)

Although many of the cases generated exhibited wide variation in number of backtracks when repeated with a randomized algorithm, none exhibited heavy tails. Figure 10 plots the number of backtracks required for a solution versus trial number, for 200 repetitions of a randomized first-fail heuristic applied to a particular problem instance with 5 attributes and 7 marginals. Figure 11 plots the running mean of the number of backtracks versus trial number, and is clearly converging.

Figure 12 shows a screen from a Java applet that conducts a systematic search for a solution to randomly generated or user-determined database consistency problem instances, for a cyclic scheme with 3 binary attributes. The URL is:

http://www.cs.sunyit.edu/faculty/mike/DB.html

Additional features (stochastic search, restarts) will be added in the very near future.

III. Using frequency database tables

A. Motivating example

Consider a scenario involving a target moving within a 3-dimensional space. Label the dimensions X, Y, and Z. We can observe the target's location in the $X-Y$ plane and $Y-Z$ plane, but not the $X-Z$ plane. However, we are interested in its location in the $X-Z$ plane (parallel to the ground). The reason for our interest is that we must decide where in the $X-Z$ plane to drop a large object (without the capability of tracking, seeking, etc.). Ideally the object will hit the target. However, if we miss, we want to avoid destroying valuable resources on the ground.

For simplicity, suppose each dimension is divided into 2 regions, left-right or up-down. Label them 0 and 1: $X=0$, $X=1$, $Y=0$, etc. (The number of regions can be made arbitrarily greater than 2 for greater realism.) Each plane is divided into 4 quadrants: $X=0$, $Y=0$; $X=0$, $Y=1$; etc. The $X-Y-Z$ cube is divided into 8 octants: $X=0$, $Y=0$, $Z=0$; etc. We will use the abbreviations $XY=01$ for $X=0$, $Y=1$, etc.

Suppose our data are in the form of frequencies of location recorded by two sensors/observers, one monitoring the $X-Y$ plane and the other monitoring the $Y-Z$ plane. The quadrant that the target is (mostly) occupying is recorded for both of the two planes at each of N equally spaced instants. For example, N=100 and at time instant 33, the target is mostly occupying quadrant $XY=01$ in the $X-Y$ plane and quadrant $YZ=11$ in the $Y-Z$ plane.

After 10,000 observations, suppose the data are:

X	Y	$f_1(.)$	Y	Z	$f_2(.)$
0	0	2000	0	0	4500
0	1	1000	0	1	500
1	0	3000	1	0	1000
1	1	4000	1	1	4000

Recall that our interest is in the $X-Z$ plane, which cannot be observed directly. What do the available data tell us about frequencies in the $X-Z$ plane? First, note that each of the 8 numbers in the frequency tables,

$$f_1(XY=00)=2000,\ f_1(XY=01)=1000,\ \ldots,\ f_2(YZ=11)=4000,$$

is a *marginal* frequency, relative to some unknown frequency distribution f_3 over X, Y and Z. For example,

$$2000 = f_1(XY=00) = f_3(XYZ=000) + f_3(XYZ=001).$$

To simplify notation, let n_i stand for $f_3(XYZ=abc)$, where i is the decimal representation of the bit string abc. Then the frequency distributions f_1 and f_2 determine a system of linear integer equations the set of solutions of which corresponds to the set of all the frequency distributions f_3 over X, Y, and Z whose *projections* onto $X-Y$ and $Y-Z$, respectively, are the distributions f_1 and f_2:

$$n_0 + n_1 = 2000$$

$$n_2 + n_3 = 1000$$

$$n_4 + n_5 = 3000$$

$$n_6 + n_7 = 4000$$

$$n_0 + n_4 = 4500$$

$$n_1 + n_5 = 500$$

$$n_2 + n_6 = 1000$$

$$n_3 + n_7 = 4000$$

(Note that the first 4 and second 4 equations each imply the constraint

$$n_0 + n_1 + \cdots + n_7 = 10,000.)$$

If there existed a single solution to this system of equations, call it

$$<n_0{*},\ n_1{*},\ \cdots,\ n_7{*}>,$$

we could calculate the values of the desired frequencies as in the table below:

X	Z	$f_4(.)$
0	0	$f_3(XYZ = 000) + f_3(XYZ = 010) = n_0 * + n_2 *$
0	1	$f_3(XYZ = 001) + f_3(XYZ = 011) = n_1 * + n_3 *$
1	0	$f_3(XYZ = 100) + f_3(XYZ = 110) = n_4 * + n_6 *$
1	1	$f_3(XYZ = 101) + f_3(XYZ = 111) = n_5 * + n_7 *$

However, there are many possible solutions. Here are three:

X	Y	Z	$f_3^1(.)$	$f_3^2(.)$	$f_3^3(.)$
0	0	0	2000	1500	1800
0	0	1	0	500	200
0	1	0	500	0	200
0	1	1	500	1000	800
1	0	0	2500	3000	2700
1	0	1	500	0	300
1	1	0	500	1000	800
1	1	1	3500	3000	3200

Suppose f_3^1 is the *actual* frequency distribution of which f_1 and f_2 are projections, and that

$$n_1 = f_3^1(XYZ = 001) = 0$$

because there is a mountain occupying most of that octant. f_3^2 is one of the solutions f_3^i of the system of equations whose value for n_1 is maximum. Each f_3^i has a corresponding relative frequency distribution rf_3^i, calculated as

$$rf_3^i(x) = f_3^i(x)/10,000.$$

The *entropy* of a relative frequency distribution rf is calculated as

$$H(rf) = -\sum_x rf(x) \times ln(rf(x)).$$

Distribution f_3^3 is the element of the solution set whose corresponding relative frequency distribution has highest entropy. Roughly, this means that it is the distribution that is closest (on various metrics) to the uniform distribution. (If the uniform distribution were a member of the solution set, it would be the maximum entropy solution.) There are a number of arguments for selecting the maximum entropy distribution from the set of distributions consistent with a given collection of constraints [11, 12]. There are also counterarguments [13, 14].

The projections of f_3^1, f_3^2 and f_3^3 onto X and Z are named, respectively, f_4^1, f_4^2 and f_4^3 in the table below:

X	Z	$f_4^1(.)$	$f_4^2(.)$	$f_4^3(.)$
0	0	2500	1500	2000
0	1	500	1500	1000
1	0	3000	4000	3500
1	1	4000	3000	3500

To add to the confusion, note that there is a second set of distributions that it would be reasonable to construct from the original data. We may choose, in effect, to ignore the information regarding the Y dimension, and project f_1 onto X and f_2 onto Z:

X	$f_5(.)$	Z	$f_6(.)$
0	3000	0	5500
1	7000	1	4500

Distributions f_5 and f_6 correspond to a system of 4 equations whose unknowns are the desired frequencies of location in the $X - Z$ plane:

$$f(XZ = 00) + f(XZ = 01) = 3000$$

$$f(XZ = 10) + f(XZ = 11) = 7000$$

$$f(XZ = 00) + f(XZ = 10) = 5500$$

$$f(XZ = 01) + f(XZ = 11) = 4500$$

The maximum entropy solution to this system of equations is:

X	Z	$f_7(.)$
0	0	1650
0	1	1350
1	0	3850
1	1	3150

The entropy of (the corresponding relative frequency distribution for) f_7 is slightly higher than that of f_4^3; thus, relative to distribution f_7, the amount of uncertainty regarding where one can expect the target to be is greater. In addition to their difference in entropy, f_7 and f_4^3 are seriously in conflict qualitatively: f_4^3 gives much greater frequency to $XZ = 00$ than to $XZ = 01$; f_7 gives greater frequency to $XZ = 10$ than to $XZ = 11$.

The distributions f_4^1, f_4^2, f_4^3 and f_7 are just 4 of astronomically many frequency distributions over XZ compatible with the available data. It will be argued later that, in a decision-making context, it is not necessary to pick a single distribution. However, the choice of *sets* of distributions will make a difference. Before introducing some formal notation and algebra, the (extremely rare) case in which one can deduce the frequencies over the attributes of interest will be illustrated.

Suppose now that the hypothetical mountain occupies most of the octants $XYZ = 001$, $XYZ = 101$, $XYZ = 000$, $XYZ = 011$. This would be consistent with the following frequency distribution:

X	Y	Z	$f_8(.)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	20
1	1	0	50
1	1	1	30

As is the case with *any* joint frequency distribution over the set of attributes $\{X, Y, Z\}$, there is a unique set of marginal distributions compatible with it (in the strict sense that, e.g., $f(XY=01) = f(XYZ=010) + f(XYZ=011)$). The converse is almost never the case (as the previous example illustrates). The marginal distributions (observed data) generated by f_8 are:

X	Y	$f_9(.)$	Y	Z	$f_{10}(.)$
0	0	0	0	0	0
0	1	0	0	1	20
1	0	20	1	0	50
1	1	80	1	1	30

From these marginal frequencies it can be deduced that the joint frequency distribution is f_8 and that the marginal frequencies of interest (over $\{X, Z\}$) are:

X	Z	$f_{11}(.)$
0	0	0
0	1	0
1	0	50
1	1	50

Distribution f_8 is *identifiable* from its projections (observed data) onto $\{X, Y\}$ and $\{Y, Z\}$. It is rare that a distribution is identifiable from projections. If the frequencies are required to sum to 100, then there are 274,436 distributions over $\{X, Y, Z\}$ that are reconstructable from their projections onto $\{X, Y\}$ and $\{Y, Z\}$. However, this is only 0.001% of the 2.6×10^{10} distributions over $\{X, Y, Z\}$ that sum to 100.

In what follows, methods are presented for dealing with the uncertainty that is virtually guaranteed to arise concerning the frequencies over the attributes of interest, when they cannot be observed directly.

B. Definitions and algebra

A *finite product space* is the Cartesian product of a finite sequence of finite sets. In what follows, the finite sets will be the *domains* of database *attributes*. For example, the discretized X dimension in the moving target example is a attribute with domain $\{0, 1\}$. As another example, the domain of the attribute Number_of_doors might be $\{1, 2, 3, 4, 5\}$. Although domains needn't be numeric, e.g., {Male, Female}, they may be encoded using integers. If A is an attribute, its domain will be denoted $dom(A)$. The finite product space associated with a set of attributes V, where $|V| = n$, is

$$S_V = \mathop{\times}_{j=1}^{n} dom(A_j),$$

where

$$<A_1, A_2, \cdots, A_n>$$

is an arbitrary (but fixed) ordering of the elements of V. The elements t of S_V are n-tuples:

$$t = < a_1, a_2, \ldots, a_n>.$$

Example:

V = {Gender, Dog_Owner}

dom(Gender) = {Male, Female}

dom(Dog_Owner) = {Yes, No}

S_V = {<Male, Yes>, <Male, No>, <Female, Yes>, <Female, No>}.

If $W \subseteq V$, the *restriction* of $t \in S_V$ to W, denoted $t[W]$, is the unique element of S_W that agrees with t on all elements of W. For example,

<Male, Yes>[{Gender}] = <Male>,

and, for $<0,0,1> \in S_{\{X,Y,Z\}}$ in the moving target example,

$$<0,0,1>[\{X, Z\}] = <0, 1>.$$

A frequency distribution over a set of attributes V is any function f from S_V to the set of nonnegative integers:

$$f: S_V \rightarrow \mathbf{N}.$$

A distribution will usually be represented in the form of a table:

Gender	Dog_Owner	$f(.)$
Male	Yes	20
Male	No	10
Female	Yes	30
Female	No	40

In the statistics literature, such tables are referred to as *contingency tables* or *cross-tabulations* [15]. As in the moving target example, the symbol f may be sub- or superscripted. The set V is the *scheme* for f, denoted $s(f)$:

$$s(f) = V.$$

A *frequency database* is a set of frequency distributions, for example, $\{f_1, f_2\}$ from the moving target example. The scheme for a database (its *database scheme*) is the set consisting of the schemes of its elements. If D is a database, $s(D)$ denotes its scheme. For example,

$$s(\{f_1, f_2\}) = \{\{X, Y\}, \{Y, Z\}\}.$$

Two algebraic operations will now be defined: *projection* and *extension*.

The *projection* of f onto W, where $W \subseteq V = s(f)$, denoted $\pi_W(f)$, is calculated as

$$\pi_W(f)(t) = \sum_{t' \in S_V | t'[W] = t} f(t').$$

The projection of a distribution onto a subset of its scheme is unique, but there may be many other distributions sharing that projection. In the moving target example,

$$f_1 = \pi_{\{X,Y\}}(f_3^1) = \pi_{\{X,Y\}}(f_3^2) = \pi_{\{X,Y\}}(f_3^3)$$

and

$$f_2 = \pi_{\{Y,Z\}}(f_3^1) = \pi_{\{Y,Z\}}(f_3^2) = \pi_{\{Y,Z\}}(f_3^3).$$

Lemma 1: If $G \subseteq H$, then $\pi_G(\pi_H(f)) = \pi_G(f)$.

An operation complementary to projection is *extension*. The extension of f to Z, where $s(f) \subseteq Z$, denoted $E^Z(f)$, is defined as

$$E^Z(f) = \{f' | f' : S_Z \to \mathbf{N},\ \pi_{s(f)}(f') = f\}.$$

For example,

$$\{f_3^1, f_3^2, f_3^3\} \subseteq E^{\{X,Y,Z\}}(f_1) \cap E^{\{X,Y,Z\}}(f_2).$$

Lemma 2: $f \in E^{s(f)}(\pi_W(f))$.

Lemma 3: $\pi_G(E^H(f)) = E^G(f)$, if $s(f) \subseteq G \subseteq H$.

For any function $g: A \to B$, the *image* of $C \subseteq A$ under g, $g(C)$, is the set of all images of elements of C:

$$g(C) = \{b | b = g(x),\ x \in C\}.$$

So, for a set of frequency distributions F, where $W \subseteq s(f)$ for all $f \in F$,

$$\pi_W(F) = \{\pi_W(f) | f \in F\}.$$

Lemmas 4 and 5, in the statement of which SF is a set of sets of frequency distributions, follow from general results in the theory of functions [16]:

Lemma 4: $\pi_W(\bigcup_{F \in SF} F) = \bigcup_{F \in SF} \pi_W(F)$.

Lemma 5: $\pi_W(\bigcap_{F \in SF} F) \subseteq \bigcap_{F \in SF} \pi_W(F)$.

The function symbols 'π' and 'E' may be overloaded, e.g., to apply to databases in addition to individual distributions or sets of distributions sharing the same scheme. Some preliminary definitions are necessary.

Database scheme S_1 is a *refinement* of scheme S_2, denoted $S_1 \leq S_2$, if every element of S_1 is a subset of some element of S_2.

Example:

$$\{\{X_1, X_2\}, \{X_2, X_3\}, \{X_3, X_4\}\} \leq \{\{X_1, X_2, X_4\}, \{X_2, X_3, X_4\}\}.$$

A distribution f may be projected onto a database scheme S if $V \subseteq s(f)$ for all $V \in S$, i.e., if $S \leq \{V\}$:

$$\pi_S(f) = \{\pi_V(f) | V \in S\}.$$

Example:

$$\pi_{\{\{X,Y\},\{Y,Z\}\}}(f_3^1) = \{f_1, f_2\}.$$

If $E^{s(f)}(\pi_S(f)) = \{f\}$, then f is *identifiable* from S.

An entire database may be projected onto a refinement of its database scheme: If $S \leq s(D)$, then

$$\pi_S(D) = \{\pi_V(f) | f \in D,\ V \in S,\ V \subseteq s(f)\}.$$

Lemma 6 is a consequence of Lemma 1.

Lemma 6: If $S_1 \leq S_2 \leq \{s(f)\}$, then

$$\pi_{S_1}(\pi_{S_2}(f)) = \pi_{S_1}(f).$$

So, in the moving target example,

$$\{f_5, f_6\} = \pi_{\{\{X\},\{Z\}\}}(\{f_1, f_2\})$$

$$= \pi_{\{\{X\},\{Z\}\}}(\pi_{\{\{X,Y\},\{Y,Z\}\}}(f_3^1))$$

$$= \pi_{\{\{X\},\{Z\}\}}(f_3^1)$$

Similarly, an entire database may be extended to a set of attributes that is a superset of the union of the elements of the database scheme:

$$E^Z(D) = \bigcap_{p \in D} E^Z(p),$$

where $\bigcup_{p \in D} s(p) \subseteq Z$.

In the target example, $E^{\{X,Y,Z\}}(\{f_1, f_2\})$ is the set of solutions f_3 to the system of 8 integer linear equations

$$f_3(<0,0,0>) + f_3(<0,0,1>) = 1500$$

$$f_3(<0,1,0>) + f_3(<0,1,1>) = 2000$$

$$\ldots$$

$$f_3(<0,1,1>) + f_3(<1,1,1>) = 2000$$

Theorem 8: Suppose that $s(D)$ is a *cover* of G, i.e., that $\bigcup_{W \in s(D)} W = G$. Suppose also that $G \subseteq H$. Then

$$\pi_G(E^H(D)) \subseteq E^G(D).$$

Proof:

$$\pi_G(E^H(D)) = \pi_G(\cap_{f \in D} E^H(f)) \qquad \text{[Definition]}$$

$$\subseteq \cap_{f \in D} \pi_G(E^H(f)) \qquad \text{[Lemma 5]}$$

$$= \cap_{f \in D} E^G(f) \qquad \text{[Lemma 3]}$$

$$= E^G(D) \qquad \text{[Definition]}$$

Theorem 9: For database schemes S_1 and S_2, $S_1 \leq S_2$ implies

$$E^Z(\pi_{S_2}(f)) \subseteq E^Z(\pi_{S_1}(f)).$$

Proof: The elements of an extension are solutions to a system of linear equations. Each of the equations determining $E^Z(\pi_{S_1}(f))$ is a linear combination of equations from the system determining $E^Z(\pi_{S_2}(f))$. Therefore, every solution to the latter system is also a solution to the former.

Returning to the moving target example, recall that we are interested in frequencies over the scheme $\{X, Z\}$. We have available the database $\{f_1, f_2\}$, with database scheme

$\{\{X, Y\}, \{Y, Z\}\}$. The database is *consistent* in the sense that

$$E^{\{X,Y,Z\}}(\{f_1, f_2\})$$

is non-empty. Thus, it is reasonable to assume that there is a distribution

$$f_3{*} \in E^{\{X,Y,Z\}}(\{f_1, f_2\})$$

such that

$$\{f_1, f_2\} = \pi_{\{\{X,Y\},\{Y,Z\}\}}(f_3{*}).$$

(This is analogous to the *universal relation assumption* in relational database theory [17].) It happens that $E^{\{X,Y,Z\}}(\{f_1, f_2\})$ is very far from empty: it is astronomically huge. Thus, $f_3{*}$ is not identifiable from its projections f_1 and f_2. If it were, then we could calculate the desired frequencies as $\pi_{\{X,Z\}}(f_3{*})$.

Assume that all of the information available regarding $f_3{*}$ is embodied in the database $\{f_1, f_2\}$. In Section III.A, the construction of two different sets of distributions (as solution sets to systems of integer linear equations), both of which contain the unknown distributions $f_3{*}$ was illustrated. One of the sets is algebraically represented as

$$\pi_{\{X,Z\}}(E^{\{X,Y,Z\}}(\{f_1, f_2\}).$$

This is the set containing f_4^1, f_4^2, and f_4^3, among many others. The other set constructed is

$$E^{\{X,Z\}}(\pi_{\{\{X\},\{Z\}\}}(\{f_1, f_2\})).$$

It contains f_7 (and many other distributions).

Although $f_3{*}$ is not identifiable, we can still make the most of the available data by deriving from it as much information as possible regarding the identity of the distribution. (Later we will discuss tradeoffs between quality of information and cost of information processing.) Which of the two approaches discussed so far retains more information about $f_3{*}$?

Theorem 10: Let $V = \bigcup_{G \in s(D)} G$. Suppose that Y is a cover of V_1, $V_0 \subseteq V_1$, and $Y \leq s(D)$. Then

$$\pi_{V_o}(E^V(D)) \subseteq \pi_{V_o}(E^{V_1}(\pi_Y(D))).$$

Proof: If $E^V(D) = \varnothing$, then

$$\pi_{V_0}(E^V(D)) = \varnothing \subseteq \pi_{V_0}(E^{V_1}(\pi_Y(D))).$$

Otherwise, there exists an f such that $D = \pi_{s(D)}(f)$. Then

$$\begin{aligned}
\pi_{V_0}(E^V(D)) &= \pi_{V_0}(E^V(\pi_{s(D)}(f))) \\
&= \pi_{V_0}(\pi_{V_1}(E^V(\pi_{s(D)}(f)))) && [\textit{Lemma } 1] \\
&\subseteq \pi_{V_0}(\pi_{V_1}(E^V(\pi_Y(f)))) && [\textit{Theorem } 9] \\
&\subseteq \pi_{V_0}(E^{V_1}(\pi_Y(f))) && [\textit{Theorem } 8] \\
&= \pi_{V_0}(E^{V_1}(\pi_Y(\pi_{s(D)}(f)))) && [\textit{Lemma } 6] \\
&= \pi_{V_0}(E^{V_1}(\pi_Y(D)))
\end{aligned}$$

Substituting $\{f_1, f_2\}$ for D, $\{X, Y, Z\}$ for V, $\{\{X\}, \{Z\}\}$ for Y, and $\{X, Z\}$ for both V_0 and V_1 in Theorem 10, we obtain

$$\pi_{\{X,Z\}}(E^{\{X,Y,Z\}}(\{f_1, f_2\}) \subseteq \pi_{\{X,Z\}}(E^{\{X,Z\}}(\pi_{\{\{X\},\{Z\}\}}(\{f_1, f_2\})))$$

$$= E^{\{X,Z\}}(\pi_{\{\{X\},\{Z\}\}}(\{f_1, f_2\})),$$

by Lemma 3.

From an information content standpoint, the set of distributions resulting from the sequence of operations extend, then project (call it EP) is superior to the set (PE) resulting from the sequence project, then extend; EP is guaranteed to be a subset of PE. For example, although we cannot identify $\pi_{\{X,Z\}}(f_3{*})$, we can infer bounds on its components. The bounds inferable from EP are tighter than the bounds inferable from PE:

X	Z	$\min_{f \in EP} f(.)$	$\max_{f \in EP} f(.)$	$\min_{f \in PE} f(.)$	$\max_{f \in PE} f(.)$
0	0	1500	3000	0	3000
0	1	0	1500	0	3000
1	0	2500	4000	2500	5500
1	1	3000	4500	1500	4500

However, to determine the tighter bounds, it is necessary to calculate maxima and minima relative to a system of 8 equations in 8 unknowns. For the looser bounds, a system of 4 equations in 4 unknowns is sufficient. Whether the additional effort is worthwhile can only be determined in the context of the use to which the information is to be put (which will include values of or estimates of other quantities, e.g., utilities), which will be discussed next.

C. Decisions with frequency tables

Recall that the motivating scenario for this research is one in which we wish to hit a target by dropping an object on it from above ($X - Z$ plane). We have recorded frequencies of occupation of quadrants in the $X - Y$ and $Y - Z$ planes. Without additional information or additional capabilities (heat seeking, etc.), the probability that we hit the target will be highest if the object is dropped into the $X - Z$ quadrant whose frequency of occupation is highest. However, we may miss the target. The costs associated with missing may vary greatly from one quadrant to another. These costs must be taken into account also.

The decision theory developed during and shortly after World War II by Von Neumann and Morgenstern [18], Savage [19] and others, is the dominant approach among researchers in artificial intelligence [20] and data fusion [21]. On this approach, when a choice is to be made among multiple actions, the action with *maximum expected utility* is selected.

On this model, the components of a decision problem are as follows:

• A set of mutually exclusive and exhaustive conditions (states)

$$C = \{c_1, c_2, \ldots, c_n\}$$

• A probability function on C,

$$p: C \to [0, 1].$$

The set of all probability functions on C is denoted P_C.

• A set of actions

$$A = \{a_1, a_2, \ldots, a_m\}$$

• A utility function

$$u: A \times C \rightarrow \mathbf{R}.$$

The elements of $A \times C$ are outcomes: $<a_1, c_j>$ is the result of performing action a_i in state c_j. We may abbreviate the pair $<a_i, c_j>$ as o_{ij}. The higher the value of $u(o_{ij})$, the more desirable is outcome o_{ij}.

Example: In the target example,

$A = \{d_{00}, d_{01}, d_{10}, d_{11}\}$, where d_{xz} stands for "drop object in quadrant xz of the $X - Z$ plane";

$C = \{l_{00}, l_{01}, l_{10}, l_{11}\}$, where l_{xz} stands for "target is located in quadrant xz of the $X - Z$ plane";

and u is given by the table

	l_{00}	l_{01}	l_{10}	l_{11}
d_{00}	0.8	0	0	0
d_{01}	0.3	1.0	0.3	0.3
d_{10}	0.5	0.5	0.9	0.5
d_{11}	0.6	0.6	0.6	1.0

(What to use for p is essentially the crux of this report and is discussed below.)

The axioms of the Von Neumann - Morgenstern theory permit assessing utility values ("utiles") on a scale from 0 to 1, assigning 1 to the most preferred outcome, 0 to the least preferred, and using introspection about hypothetical lotteries to assign values for the other outcomes. Outcomes $<d_{01}, l_{01}>$ and $<d_{11}, l_{11}>$ are tied for most preferred because region l_{01} is a mountain, l_{11} is water, l_{00} is a city, and l_{10} is sparsely populated. Similarly, since l_{00} is a city, outcomes $<d_{00}, l_{01}>$, $<d_{00}, l_{10}>$, and $<d_{00}, l_{11}>$ are tied for worst. To assess the utility of $<d_{01}, l_{10}>$, for example, a domain expert is questioned by a decision analyst (the process can be and has been automated) regarding preferences between $<d_{01}, l_{10}>$ for certain and a lottery with two "prizes": $<d_{01}, l_{01}>$ (one of the most preferred outcomes) and $<d_{00}, l_{11}>$ (one of the least preferred). Initially, the probabilities of "winning" each prize are equal: 0.5. If a preference for $<d_{01}, l_{10}>$ is stated, then the lottery is changed to give probability 0.75 to $<d_{01}, l_{01}>$ (making the new lottery more desirable than the old one). If now the lottery is preferred, the probability of $<d_{01}, l_{01}>$ is reduced to 0.625. (The probabilities are usually represented graphically, as pie charts or "wheels of fortune".) This continues until the domain expert expresses indifference. If p is the final probability for $<d_{01}, l_{01}>$, then it follows from the utility axioms that

$$u(< d_{01}, l_{10} >) = p \times u(< d_{01}, l_{01} >) + (1 - p) \times u(< d_{00}, l_{11} >)$$

$$= p \times 1.0 + (1 - p) \times 0$$

$$= p.$$

The *expected* utility of an action a_i is computed as

$$EU(a_i) = \sum_{j=1}^{n} p(c_j | a_i) \times u(< a_i, c_j >).$$

Any decision problem for which the performance of an action affects the probability of a condition, i.e., for which

$$p(c_j|a_i) \neq p(c_j)$$

can be transformed into an equivalent problem in which the actions do not affect the probabilities of conditions [22]. In what follows, we will assume that our decision problems have the property

$$p(c_j|a_i) = p(c_j),$$

i.e., that

$$EU(a_i) = \sum_{j=1}^{n} p(c_j) \times u(< a_i, c_j >)).$$

For our moving target example, we need probabilities for the conditions l_{xz}. Let us suppose, again, that the available data are the frequency distributions f_1 and f_2, each of which is the result of 10,000 (coordinated) observations:

X	Y	$f_1(.)$	Y	Z	$f_2(.)$
0	0	2000	0	0	4500
0	1	1000	0	1	500
1	0	3000	1	0	1000
1	1	4000	1	1	4000

So far, two methods of obtaining frequencies over the attributes $\{X, Z\}$ of interest have been discussed: Pick an element of

$$\pi_{\{X,Z\}}(E^{\{X,Y,Z\}}(\{f_1, f_2\})),$$

or pick an element of

$$E^{\{X,Z\}}(\pi_{\{\{X\},\{Z\}\}}(\{f_1, f_2\})).$$

We have identified 3 elements (out of astronomically many altogether) of the former set of distributions, f_4^1, f_4^2, and f_4^3; and one element (out of astronomically many) of the latter, f_7. One way to obtain probabilities from these frequencies is to convert them to relative frequencies. Let p_i^j denote the relative frequency distribution associated with f_i^j:

$$p_i^j(t^0) = f_i^j(t^0)/\sum_t f_i^j(t).$$

Since, for this example, the total number of observations (over a sample space of four elementary events) is 10,000, we could be quite confident in any one of the p_i^j , regarded in isolation, as an estimate of the unknown probability of occupation of quadrants in the $X - Z$ plane. 95% confidence intervals for p_4^1, for example, are very tight [23]:

X	Z	$f_4^1(.)$	$p_4^1(.)$	95% conf. limits
0	0	2500	0.25	+/-0.004
0	1	500	0.05	+/-0.002
1	0	3000	0.30	+/-0.005
1	1	4000	0.40	+/-0.005

Definition: An *interval-valued* probability distribution over a set of attributes V is a mapping

$$i: S_V \to I,$$

where I is the set of closed, convex subsets (subintervals) of [0, 1], $\sum_{t \in S_V} \max i(t) \geq 1$, $0 \leq \sum_{t \in S_V} \min i(t) \leq 1$. We may then associate the interval distribution i_4^1 with the 95%

confidence limits on the values of p_4^1:

X	Z	$i_4^1(.)$
0	0	[0.246, 0.254]
0	1	[0.048, 0.052]
1	0	[0.295, 0.305]
1	1	[0.395, 0.405]

We may do the same for distributions f_4^2, f_4^3 and f_7:

X	Z	i_4^2	i_4^3	i_7
0	0	[0.146, 0.154]	[0.196, 0.204]	[0.161, 0.169]
0	1	[0.146, 0.154]	[0.097, 0.103]	[0.132, 0.138]
1	0	[0.395, 0.405]	[0.345, 0.355]	[0.380, 0.390]
1	1	[0.295, 0.305]	[0.345, 0.355]	[0.310, 0.320]

These probability estimates are in serious conflict: not only are the 4-way intersections

$$i_4^1(t) \cap i_4^2(t) \cap i_4^3(t) \cap i_7(t)$$

empty for all t, so are each of the 6 2-way intersections – no interval distribution agrees with any other on any of the 4 values $t \in S_{\{X,Z\}}$.

Definition: $i_1 \colon S_V \to I$ and $i_2 \colon S_V \to I$ *agree on* $t \in S_V$ if

$$i_1(t) \cap i_2(t) \neq \varnothing.$$

i_1 and i_2 *agree* if they agree on all $t \in S_V$.

The expected utilities of the actions d_{00}, d_{01}, d_{10}, and d_{11}, relative to each of the probability distributions p_4^1, p_4^2, p_4^3, and p_7 are given in the table below:

	p_4^1	p_4^2	p_4^3	p_7
d_{00}	0.200	0.120	0.160	0.132
d_{01}	0.335	0.206	0.171	0.395
d_{10}	0.620	0.660	0.640	0.654
d_{11}	0.760	0.412	0.740	0.726

There is less conflict among the distributions with respect to the action identified as optimal. Although the expected utilities for a given action relative to the four distributions considered vary widely, three of the four give maximum expected utility to action d_{11}. This leads to the concept of agreement among distributions in the context of a particular decision-making scenario.

The *domain* of an action a_i, denoted $D(a_i)$, is the set of probability distributions over C relative to which the expected utility of a_i is maximum (or tied for maximum) among all actions under consideration:

$$D(a_i) = \{\, p \in P_C \mid \sum_{j=1}^{n} p(c_j) \times u(< a_i, c_j >) \geq \sum_{j=1}^{n} p(c_j) \times u(< a, c_j >), \text{ for all } a \in A \,\}.$$

To illustrate, the domain of action d_{11}, $D(d_{11})$, is the set of all solutions p to the system of linear equations:

$$p(l_{00}) + p(l_{01}) + p(l_{10}) + p(l_{11}) = 1$$

$$0.6p(l_{00}) + 0.6p(l_{01}) + 0.6p(l_{10}) + p(l_{11}) \geq 0.8p(l_{00})$$

$$0.6p(l_{00}) + 0.6p(l_{01}) + 0.6p(l_{10}) + p(l_{11}) \geq 0.3p(l_{00}) + p(l_{01}) + 0.3p(l_{10}) + 0.3p(l_{11})$$

$$0.6p(l_{00}) + 0.6p(l_{01}) + 0.6p(l_{10}) + p(l_{11}) \geq 0.5p(l_{00}) + 0.5p(l_{01}) + 0.9p(l_{10}) + 0.5p(l_{11})$$

Distributions p_4^1, p_4^3, and p_7 are elements of $D(d_{11})$; p_4^2 is not.

Any decision problem (with components C, A, and u) decomposes P_C into convex subsets $D(a_1), D(a_2), \cdots, D(a_m)$:

$$P_C = \bigcup_{i=1}^{m} D(a_i).$$

The decomposition is not a partition, however, since the domains (convex polyhedra) can share faces.

In the context of a decision problem, distributions p and p' are *equivalent* if they belong to the same domain(s). Thus, modulo this problem, despite diverging dramatically (e.g., $p_4^1(l_{01}) = 0.050$, $p_7(l_{01}) = 0.135$), p_4^1, p_4^3 and p_7 are equivalent. Note, though, that p_4^1, p_4^2, p_4^3 and p_7 are just 4 of infinitely many distributions over $\{X, Z\}$ that are compatible in some sense with the original frequency data. That 3 of these 4 belong to $D(d_{11})$ is not sufficient reason to perform action d_{11}.

D. Data Consistency

It is assumed in the moving target example that the observations are coordinated, accurate, and correctly recorded. As a result, the database is consistent. A database D is *globally consistent* if

$$E^V(D) \neq \varnothing,$$

where $V = \bigcup_{W \in s(D)} W$. A database $D = \{f_1, f_2, \cdots f_k\}$ is *locally consistent* if

$$\pi_{s(f_i)\cap s(f_j)}(f_i) = \pi_{s(f_i)\cap s(f_j)}(f_j),$$

for all $f_i, f_j \in D$. (The distributions are said to be *pairwise consistent.*)

If $s(D)$, viewed as a hypergraph, is α-acyclic [10], then if D is locally consistent, it is also globally consistent. It is computationally cheaper to test for pairwise (local) consistency of the elements of D than to check for a feasible solution to the system of integer linear equations corresponding to $E(D)$. (The latter is an NP-complete problem [2].)

To test for α-acyclicity, apply step (0) to $s(D)$, then steps (1) and (2), in any order, until neither has any effect on G:

(0) $G = s(D)$

(1) If v is contained in only one $V \in G$, $V = V - \{v_1\}$

(2) If $V_1 \in G$ and $V_2 \in G$ and $V_1 \subseteq V_2$, $G = G - \{V_1\}$

If the final value of G is $\{\varnothing\}$, then $s(D)$ is α-acyclic; otherwise, $s(D)$ is α-cyclic [10].

Example: $\{\{X, Y\}, \{Y, Z\}\}$ is α-acyclic:

(a) $\{\{X, Y\}, \{Y, Z\}\}$

(b) $\{\{Y\}, \{Y, Z\}\}$

(c) $\{\{Y, Z\}\}$

(d) $\{\{Z\}\}$

(e) $\{\varnothing\}$

Example: $\{\{X, Y\}, \{Y, Z\}, \{X, Z\}\}$ is α-cyclic:

(a) $\{\{X, Y\}, \{Y, Z\}, \{X, Z\}\}$

Neither step (1) nor step (2) is applicable.

Since $\{\{X, Y\}, \{Y, Z\}\}$ is α-acyclic, the moving target database can be tested for (global) consistency by testing local consistency:

Y	$\pi_{\{Y\}}(f_1)(.)$	$\pi_{\{Y\}}(f_2)(.)$
0	$2000 + 3000 = 5000$	$5000 = 4500 + 500$
1	$1000 + 4000 = 5000$	$5000 = 1000 + 4000$

The algebra presented so far applies also to the relative frequency distributions p derivable from frequency distributions f as

$$p(t) = f(t) / \sum_t f(t).$$

The same basic arithmetic operation, addition, underlies the operations of projection and extension for frequencies and probabilities. An additional operation, *join*, involving multiplication and real-number division, is meaningful for probability distributions. The join of two distributions p_1 and p_2 is calculated as

$$J(p_1, p_2)(t) = p_1(a) \times p_2(b) / \sum_{c | c[s(p_1) \cap s(p_2)] = b[s(p_1) \cap s(p_2)]} p_2(c),$$

where $a = t[s(p_1)]$, $b = t[s(p_2)]$, and where the denominator equals 1 if $s(p_1) \cap s(p_2) = \varnothing$.

Calculating $p(t)$ in this way amounts to the assumption of *conditional independence* of $s(p_1) - (s(p_1) \cap s(p_2))$ and $s(p_2) - (s(p_1) \cap s(p_2))$, given $s(p_1) \cap s(p_2)$ [20]. $J(p_1, p_2)$ is also the maximum entropy element of $E^{s(p_1) \cup s(p_2)}(\{p_1, p_2\})$ [24].

Example: Let p_1 and p_2 be the relative frequency distributions associated with f_1 and f_2 from the moving target example. Then $J(p_1, p_2)$ is calculated as in the table below:

X	Y	Z	$J(p_1, p_2)(t)$
0	0	0	$0.18 = 0.20 \times 0.45/(0.45 + 0.05)$
0	0	1	$0.02 = 0.20 \times 0.05/(0.45 + 0.05)$
0	1	0	$0.02 = 0.10 \times 0.10/(0.10 + 0.40)$
0	1	1	$0.08 = 0.10 \times 0.40/(0.10 + 0.40)$
1	0	0	$0.27 = 0.30 \times 0.45/(0.45 + 0.05)$
1	0	1	$0.03 = 0.30 \times 0.05/(0.45 + 0.05)$
1	1	0	$0.08 = 0.40 \times 0.10/(0.10 + 0.40)$
1	1	1	$0.32 = 0.40 \times 0.40/(0.10 + 0.40)$

If $s(D)$, where $D = \{f_1, f_2, \cdots, f_k\}$, is α-acyclic, then the maximum entropy element of $E(\{p_1, p_2, \cdots, p_k\})$ can be calculated by a sequence of applications of the join operation: $J(p_{\sigma(k)}, J(p_{\sigma(k-1)}, \cdots, J(p_{\sigma(2)}, p_{\sigma(1)}) \cdots))$, where $s(p_{\sigma(1)})$ is the first element of $s(D)$ eliminated by an application of the test for acyclicity, $s(p_{\sigma(2)})$ is the second, etc. If $s(D)$ is α-cyclic, calculation of the maximum entropy element of $E(\{p_1, p_2, \cdots, p_k\})$ requires application of an expensive *iterative proportional fitting* algorithm [25].

Thus, there are computational advantages to databases with α-acyclic schemes. On the other hand, there are informational advantages to cyclic schemes.

Theorem 11: For any distribution f and any database $D = \pi_{s(D)}(f)$ over set of attributes $s(f)$, where $s(D) \neq \{s(f)\}$ and $|s(f)| \geq 3$, if $s(D)$ is α-acyclic, then there exists a database D' over $s(f)$ such that

(1) $s(D')$ is α-cyclic;

(2) $D' = \pi_{s(D')}(f)$;

and

(3) $E^{s(f)}(D') \subseteq E^{s(f)}(D)$.

Proof: First, note that if $s(D) = \{s(f)\}$, then $D = \{f\}$. If $s(D) \neq \{s(f)\}$ and $s(D)$ is α-acyclic, let

$$D' = \{K | K \subseteq s(f) \text{ and } |K| = |s(f)| - 1\}.$$

D' is α-cyclic: Each attribute is contained in $|s(f)| - 1$ elements of D'. Since $|s(f)| \geq 3$, $|s(f)| - 1 \geq 2$, and no attribute is contained in only one element of D'; so step (1) of the cyclicity test algorithm is not applicable. Since no element of D' is a subset of any other element of D', step (2) is also inapplicable. D is a refinement of D': for every element $V \in D$ there is a $V' \in D'$ such that $V \subseteq V'$. Therefore, by Theorem 9,

$$E^{s(f)}(\pi_{s(D')}(f)) \subseteq E^{s(f)}(\pi_{s(D)}(f)),$$

which implies

$$E^{s(f)}(D') \subseteq E^{s(f)}(D).$$

Example: Suppose the actual frequency distribution (over 100 time units) is

X	Y	Z	$f_{12}(.)$
0	0	0	50
0	0	1	0
0	1	0	20
0	1	1	0
1	0	0	0
1	1	0	5
1	1	1	25

If the available data are $\pi_{\{X,Y\}}(f_{12})$ and $\pi_{\{Y,Z\}}(f_{12})$, the inferable bounds on f_{12} are

X	Y	Z	min	max
0	0	0	45	50
0	0	1	0	5
0	1	0	0	20
0	1	1	0	20
1	0	0	0	5
1	1	0	5	20
1	1	1	5	25

However, f_{12} is identifiable from its projections onto $\{\{X, Y\}, \{Y, Z\}, \{X, Z\}\}$. On the other hand, although observations over the scheme $\{\{X, Y\}, \{Y, Z\}, \{X, Z\}\}$ provide more information, obtaining the additional data may be too expensive, too dangerous, or impossible. Furthermore, a system of 12 equations is required for the less refined scheme, vs. 8 equations for the more refined scheme.

It is quite likely that frequencies obtained experimentally vs. by projection from a joint frequency distribution over any scheme that is not a partition, even if it is α-acyclic, will be incosistent.

Example: Suppose that coin c_2 is tossed 250 times in conjunction with coin c_1 and another 250 times in conjunction with coin c_3, and the joint toss outcomes are recorded in the table below.

c_1	c_2	$f_{13}(.)$	c_2	c_3	$f_{14}(.)$
H	H	70	H	H	65
H	T	60	H	T	50
T	H	40	T	H	70
T	T	80	T	T	65

The database $\{f_{13}, f_{14}\}$ is inconsistent, as is the database of relative frequencies

c_1	c_2	$p_{13}(.)$	c_2	c_3	$p_{14}(.)$
H	H	0.28	H	H	0.26
H	T	0.24	H	T	0.20
T	H	0.16	T	H	0.28
T	T	0.32	T	T	0.26

Distributions p_{13} and p_{14} give conflicting information on the bias of coin c_2:

$$\pi_{\{c_2\}}(p_{13}) \neq \pi_{\{c_2\}}(p_{14}).$$

However, standard 95% confidence intervals for p_{13} and p_{14} are consistent:

c_1	c_2	$i_{13}(.)$	c_2	c_3	$i_{14}(.)$
H	H	[0.224, 0.336]	H	H	[0.206, 0.314]
H	T	[0.187, 0.293]	H	T	[0.150, 0.250]
T	H	[0.115, 0.205]	T	H	[0.224, 0.336]
T	T	[0.262, 0.378]	T	T	[0.206, 0.314]

A pair of interval distributions i and i' is *consistent* if

$$\pi_{s(i)\cap s(i')}(i)(t) \cap \pi_{s(i)\cap s(i')}(i')(t) \neq \varnothing,$$

for all t over $s(i) \cap s(i')$. (Note that this is a more general concept than *agreement*, as defined in Section III.C.) The projection of an interval distribution is obtained via linear programming. For example, the endpoints of

$$\pi_{\{c_2\}}(i_{13})(H)$$

are obtained by minimizing and maximizing $p(H)$ subject to the constraints

$$p(H) = p(HH) + p(HT)$$

$$p(T) = p(TH) + p(TT)$$

$$p(H) + p(T) = 1$$

$$p(HH) \geq 0.224$$

$$p(HH) \leq 0.336$$

$$\cdot$$

$$\cdot$$

$$\cdot$$

$$p(TT) \geq 0.262$$

$$p(TT) \leq 0.378$$

The remaining endpoints are obtained similarly:

c_2	$\pi_{\{c_2\}}(i_{13})(.)$	$\pi_{\{c_2\}}(i_{14})(.)$
H	[0.339, 0.541]	[0.356, 0.564]
T	[0.459, 0.661]	[0.436, 0.644]

(As this example illustrates, the endpoints for a projection are not always sums of endpoints of the projected intervals.) Since the intersections are all non-empty, i_{13} and i_{14} are consistent.

The (real-valued) *extension* of an interval database $I = \{i_1, \cdots, i_k\}$, $E^V(I)$, is the set of all (real-valued) probability distributions over V whose (real-valued) projections onto $s(I)$ satisfy the inequality constraints implied by the elements of I.

Example: $E^{\{c_1,c_2,c_3\}}(\{i_{13}, i_{14}\})$ is the set of all solutions p to the system of inequalities:

$$p(HHH) + p(HHT) \geq 0.224$$

$$p(HHH) + p(HHT) \leq 0.336$$

$$\cdot$$

$$\cdot$$

$$p(HTT) + p(TTT) \geq 0.206$$

$$p(HTT) + p(TTT) \leq 0.314$$

$$p(HHH) + \cdots + p(TTT) = 1$$

An interval database I is globally consistent if $E^V(I) \neq \emptyset$, where $V = \bigcup_{i \in I} s(i)$. If $s(I)$ is α-acyclic, local consistency of I is sufficient for global consistency.

V. Decisions with interval databases.

Frequency and relative frequency databases are likely to be inconsistent. We may replace them with interval databases, which are much less likely to be inconsistent. We may then work with probability intervals or sets of (real-valued) probability distributions in selecting an optimal action. This requires adaptation of the classical maximum expected utility criterion.

In the moving target example, suppose we have far fewer than 10,000 observations and that the database is inconsistent (due perhaps to differences in perspective, when the object is not entirely within a single $X - Y - Z$ cell):

X	Y	$f_{15}(.)$	Y	Z	$f_{16}(.)$
0	0	10	0	0	25
0	1	25	0	1	5
1	0	15	1	0	60
1	1	50	1	1	10

95% intervals:

X	Y	$i_{15}^{0.95}(.)$	Y	Z	$i_{16}^{0.95}(.)$
0	0	[0.041, 0.159]	0	0	[0.165, 0.335]
0	1	[0.165, 0.335]	0	1	[0.007, 0.093]
1	0	[0.106, 0.194]	1	0	[0.539, 0.661]
1	1	[0.402, 0.598]	1	1	[0.041, 0.159]

We may project $E^{\{X,Y,Z\}}(\{i_{15}^{0.95}, i_{16}^{0.95}\})$ onto $\{X, Z\}$ via linear programming to construct the interval distribution:

X	Z	$i_{17}(.)$
0	0	[0.006, 0.492]
0	1	[0.000, 0.252]
1	0	[0.256, 0.792]
1	1	[0.000, 0.252]

With the set of actions, conditions and utility function of Section III.C, these probability intervals determine expected utility intervals:

$$EU(d_{00}) \in [0.0048, 0.3936]$$
$$EU(d_{01}) \in [0.3000, 0.4764]$$
$$EU(d_{10}) \in [0.6024, 0.8168]$$
$$EU(d_{11}) \in [0.6000, 0.7008]$$

It is now possible to eliminate actions d_{00} and d_{01} from consideration: they are both *dominated*, in the sense that there exist actions (d_{10} and d_{11}) whose minimum expected utility is greater than their maximum expected utility.

We need to choose between d_{10} and d_{11}. If additional sampling is possible, it will narrow the probability intervals, which in turn will narrow the utility intervals, increasing the likelihood that a single action will be undominated.

If additional sampling is not possible, we may narrow the intervals by choosing a lower confidence level. This gives a "context-driven" solution to how conservatively we should treat our observed relative frequencies as estimates of probabilities. A second alternative is to base the choice between the undominated actions on their *security levels* [26]: choose the (undominated) action whose minimum utility is maximum ("maximin"). This policy would select d_{11} (as it would if we had no frequency data whatsoever).

The algebraic results in Section III.B permit additional strategies that are useful when dealing with larger numbers of attributes. In a manner analogous to iterative deepening search, we may project the database onto a sequence of increasingly less refined schemes, starting with

$$\{\{v\} | v \in V\},$$

where V is the set of attributes directly relevant to the decision problem. It follows from Theorem 10 that the utility intervals will become narrower at each step in the sequence. However, the number of variables and inequalities in the corresponding linear program increases also with each iteration.

REFERENCES

[1] T. Hogg *et al.*, "Phase transitions and the search problem." *Artificial Intelligence, v. 81*, 1996, pp. 1-15.

[2] M. Garey and D. Johnson. *Computers and Intractability.* W. H. Freeman, San Francisco, 1979.

[3] B. Selman *et al.*, "Generating hard satisfiability problems." *Artificial Intelligence, v. 81*, 1996, pp. 17-29.

[4] C. Gomes and B. Selman, "Problem structure in the presence of perturbations." *Proc. of the 13th Conf. on Uncertainty in Artificial Intelligence*, 1997.

[5] S. Reiss, "Statistical database confidentiality." Report No. 25, Dept. of Statistics, Univ. of Stockholm, 1977.

[6] D. Brelaz, "New methods to color the vertices of a graph." *Communications of the ACM, v. 22*, 1979, pp. 251-256.

[7] ILOG. URL = http://www.ilog.com/

[8] I. Gent *et al.*, "The constrainedness of search." *AAAI-96*, pp. 246-252.

[9] C. Gomes *et al.*, "Heavy-tailed probability distributions in combinatorial search." In:

Principles and Practice of Constraint Programming, G. Smoka, Ed., Springer-Verlag, 1997, pp. 121-135.

[10] R. Fagin. "Degrees of acyclicity for hypergraphs and relational database schemes." *J. of the ACM, v. 30*, n. 3, pp. 514-550, 1983.

[11] E. T. Jaynes. "On the rationale of maximum-entropy methods." *Proc. of the IEEE, v. 70*, n. 9, pp. 939-952, 1982.

[12] J. Shore and R. Johnson. "Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy." *IEEE Trans. Information Theory, v 26*, n. 1, pp 26-37, 1980.

[13] T. Seidenfeld. "Entropy and uncertainty." *Philosophy of Science, v. 53*, pp. 467-491, 1986.

[14] H. Kyburg and M. Pittarelli. "Set-based Bayesianism." *IEEE Trans. Systems, Man, and Cybernetics, v. 26*, n. 3, pp. 324-339, 1996.

[15] Y. M. M. Bishop, S. Fienberg, and P. Holland. *Discrete Multivariate Analysis*. MIT Press, Cambridge, MA, 1975.

[16] N. Bourbaki. *Theory of Sets*. Addison-Wesley, Reading, MA, 1968.

[17] D. Maier. *The Theory of Relational Databases*. Computer Science Press, Rockville, MD, 1983.

[18] J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton Univ. Press, Princeton, NJ, 1944.

[19] L. J. Savage. *The Foundations of Statistics*. Wiley, New York, 1954.

[20] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Francisco, 1988.

[21] E. Waltz and J. Llinas. *Multisensor Data Fusion*. Artech House, Norwood, MA, 1990.

[22] R. Jeffrey. "Savage's omelet." In: *PSA 1976, v. 2*, F. Suppe and P. Asquith, Eds., Philosophy of Science Association, E. Lansing, MI, 1976.

[23] W. Cochran. *Sampling Techniques*. Wiley, New York, 1977.

[24] P. M. Lewis. "Approximating probability distributions to reduce storage requirements." *Information and Control, v. 2*, pp. 214-225, 1959.

[25] K. Krippendorff. *Information Theory: Structural Models for Qualitative Data*. Sage, Beverly Hills, CA, 1986.

[26] I. Levi. *The Enterprise of Knowledge*. MIT Press, Cambridge, MA, 1980.

FIGURES

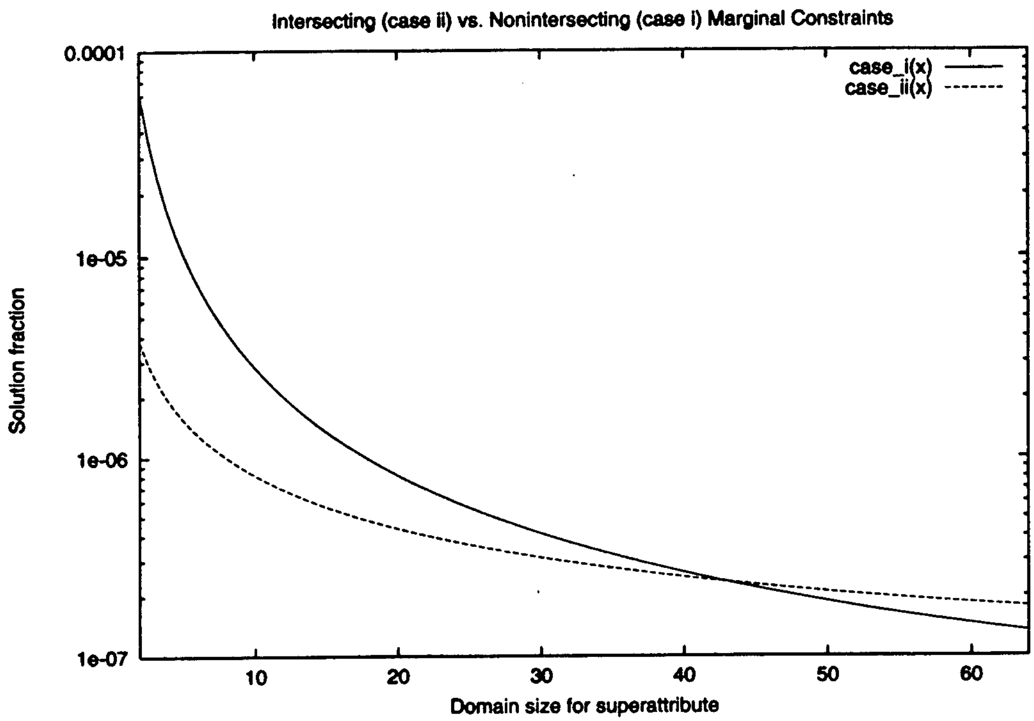


Figure 1

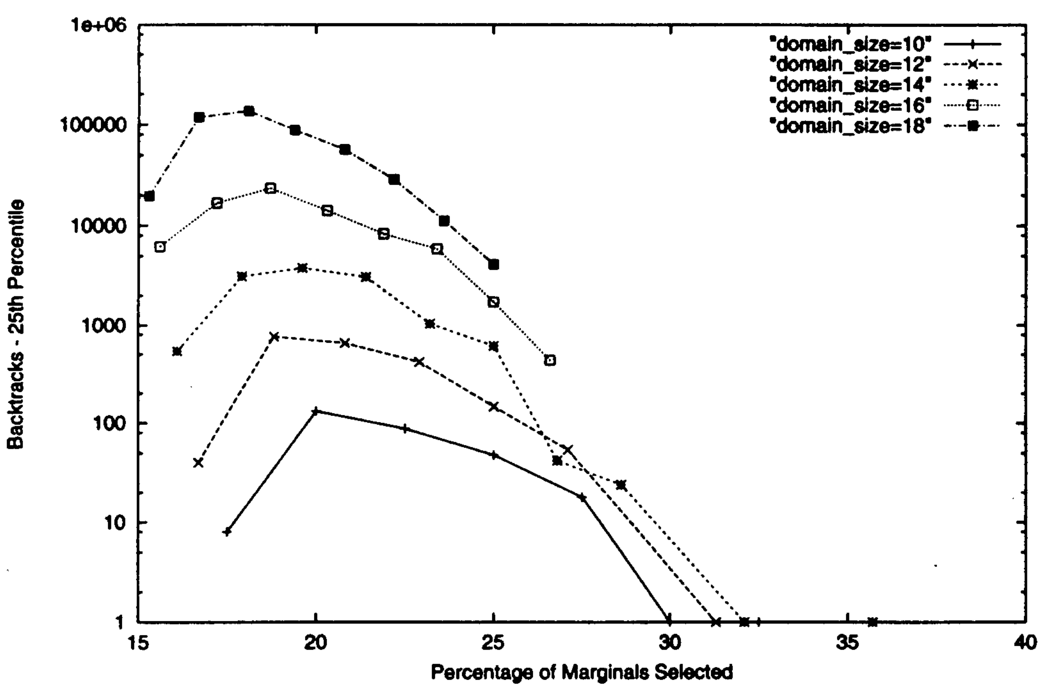


Figure 2

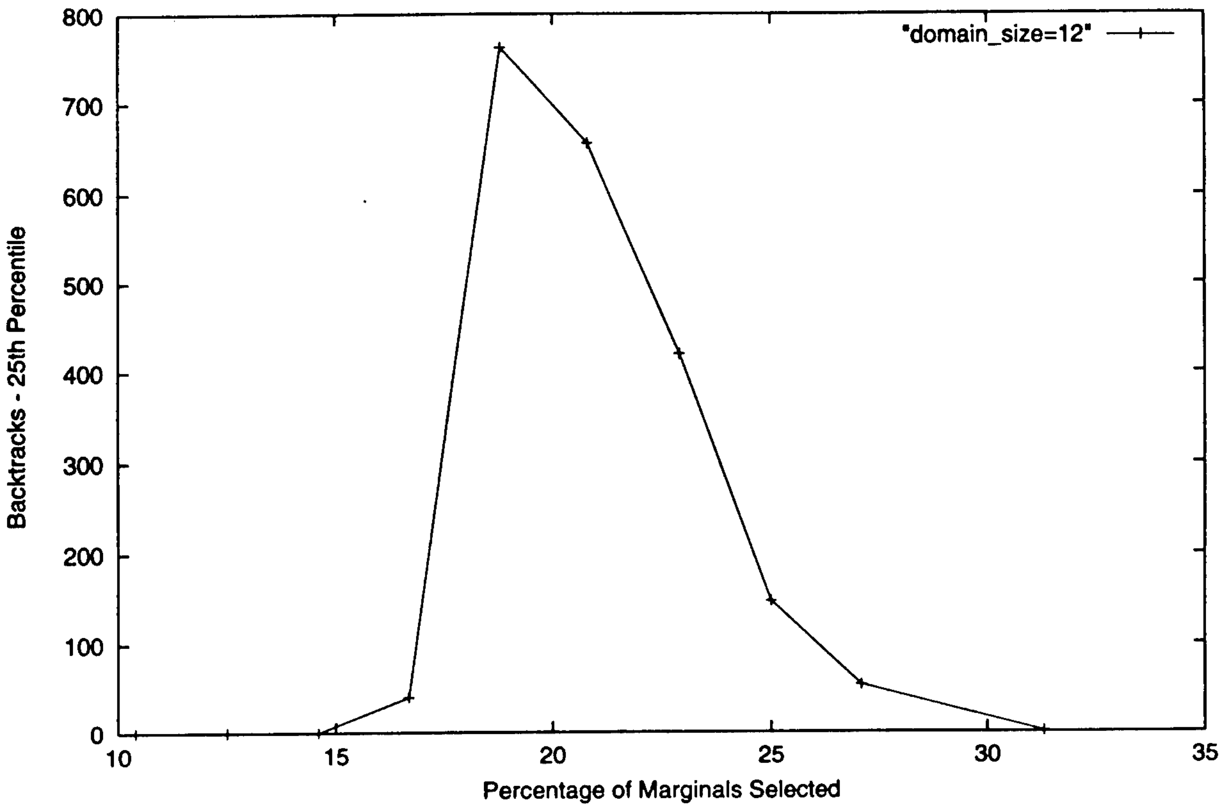


Figure 3

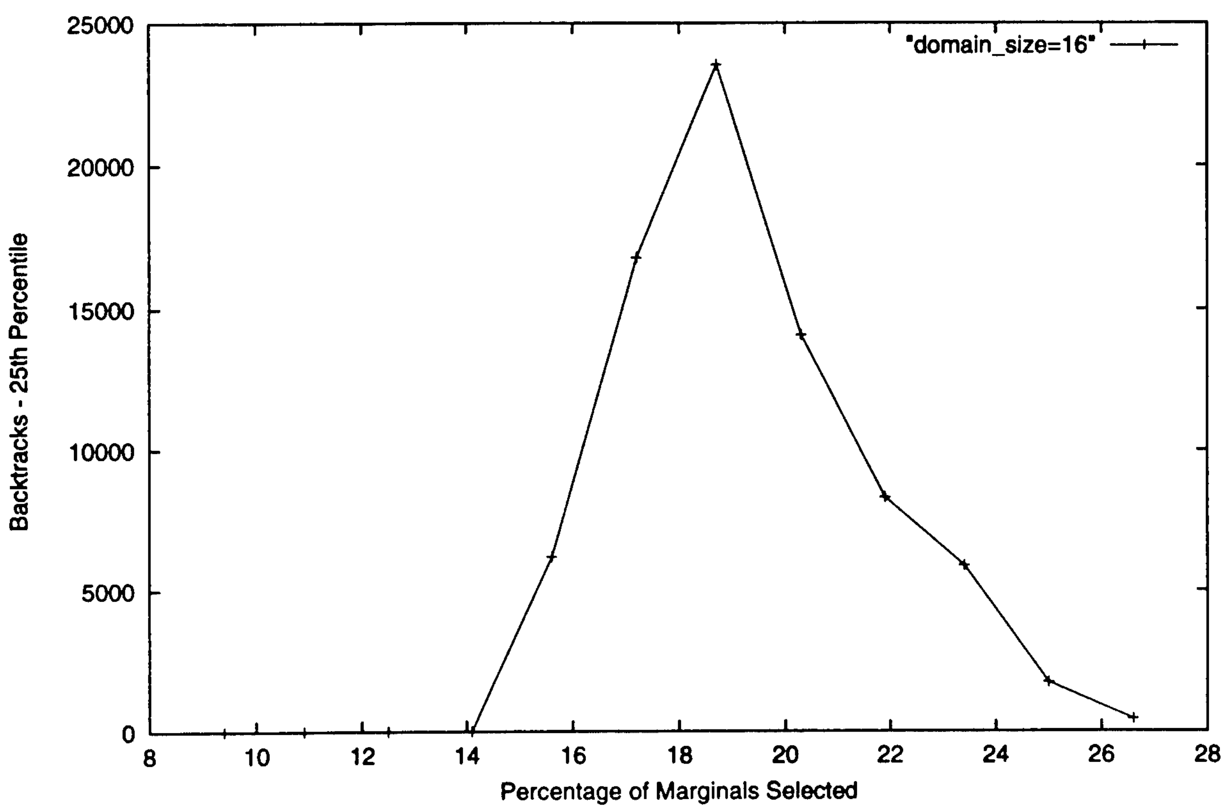


Figure 4

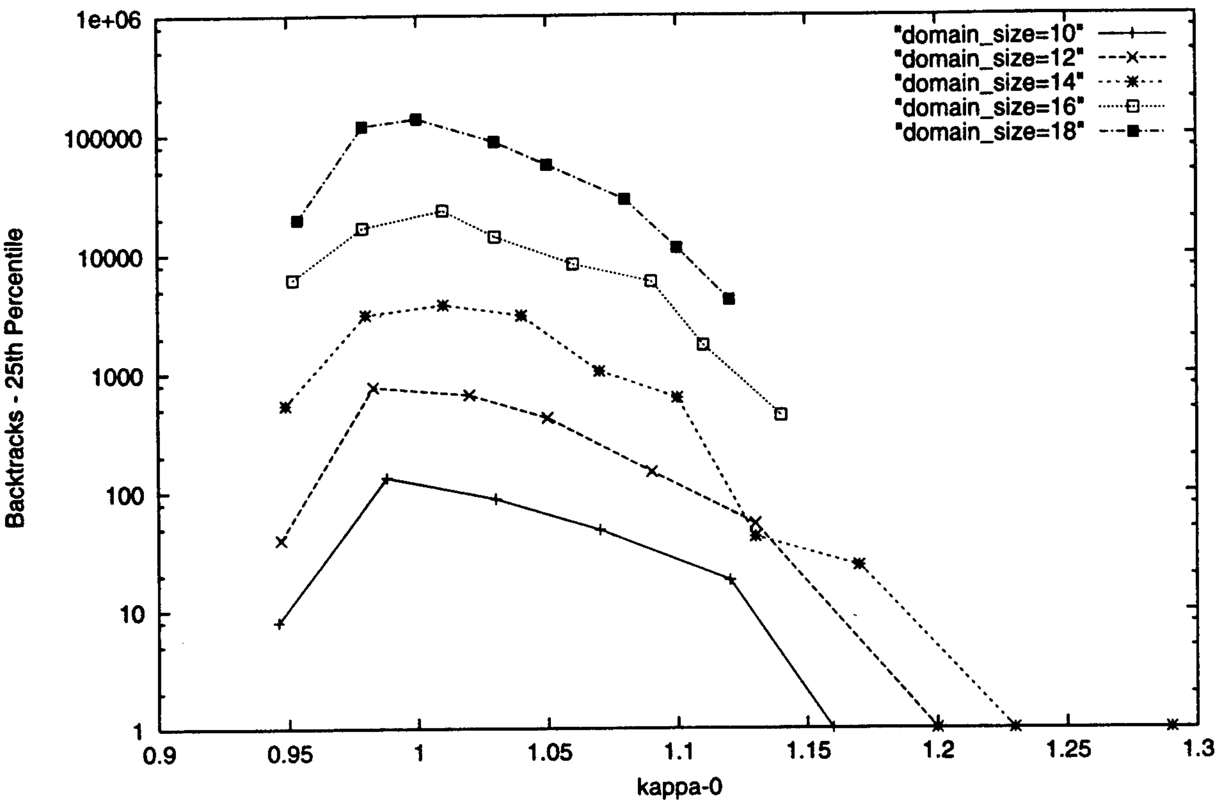


Figure 5

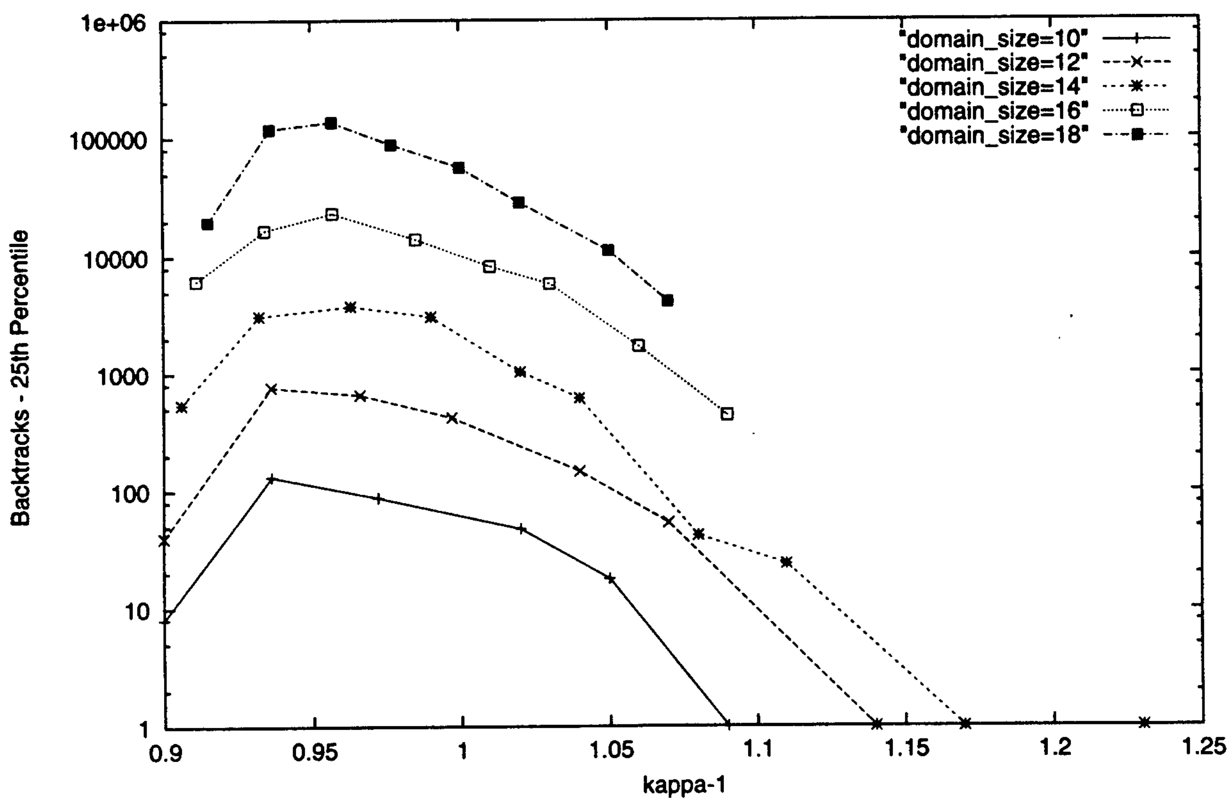


Figure 6

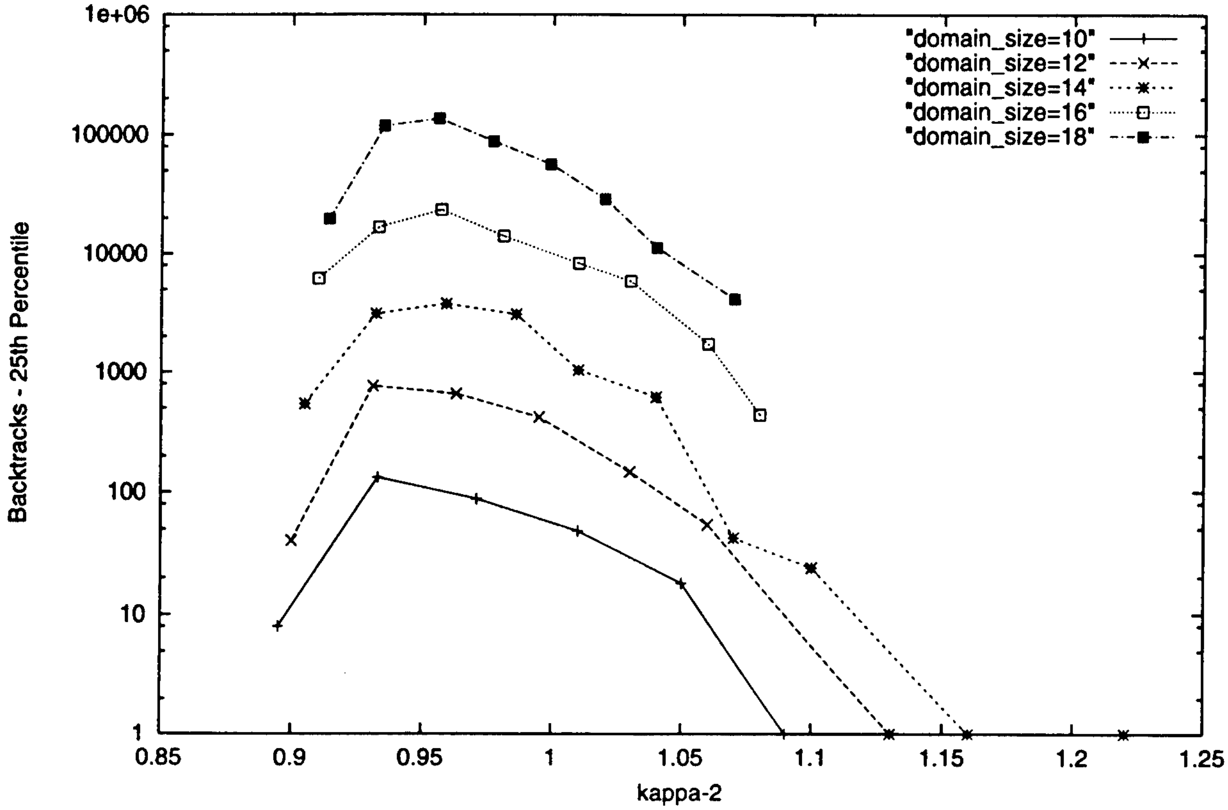


Figure 7

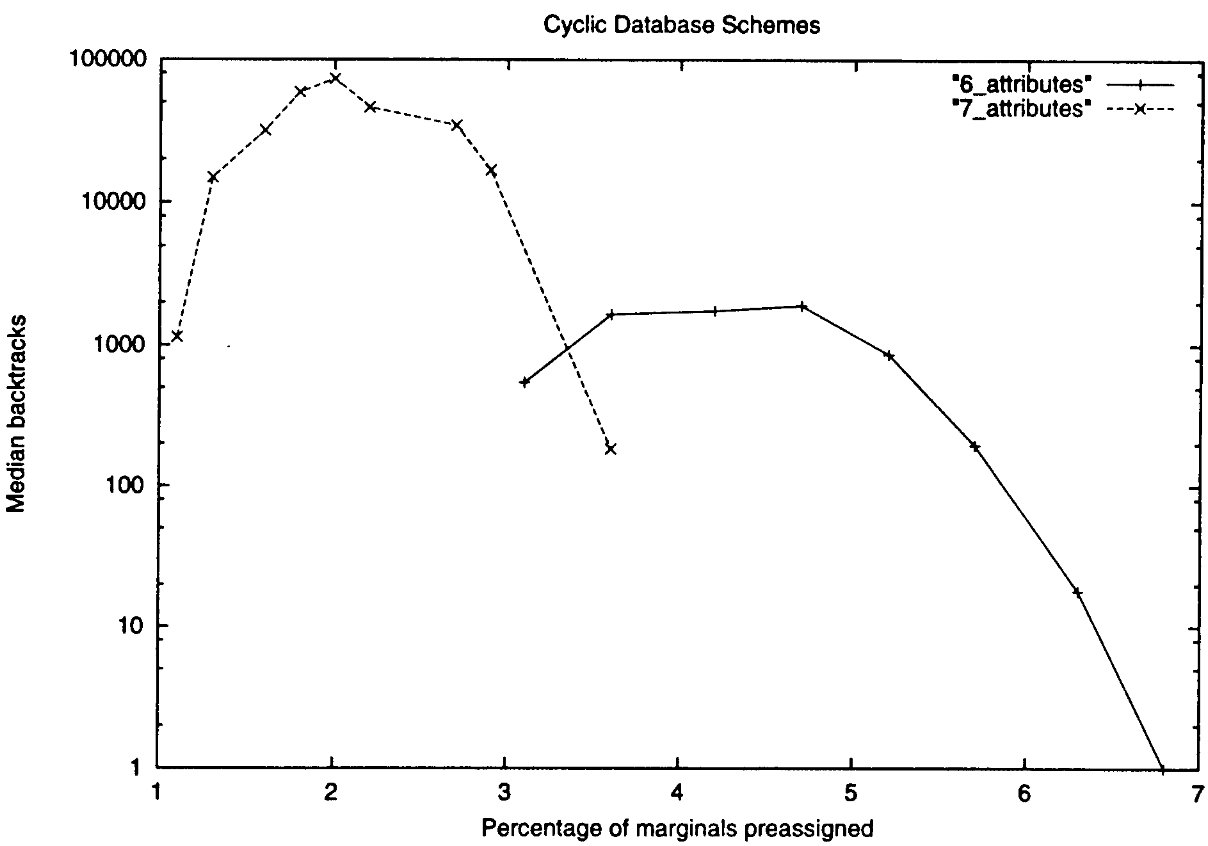


Figure 8

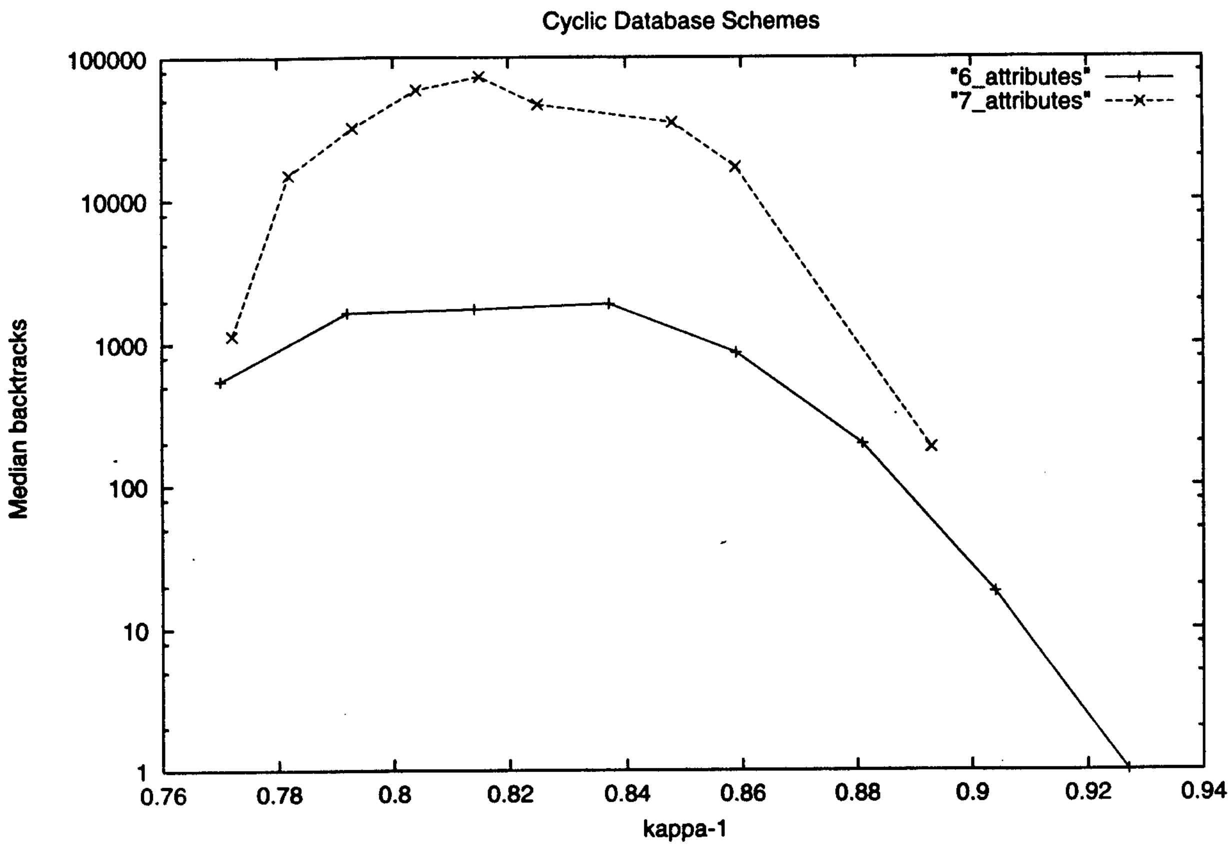


Figure 9

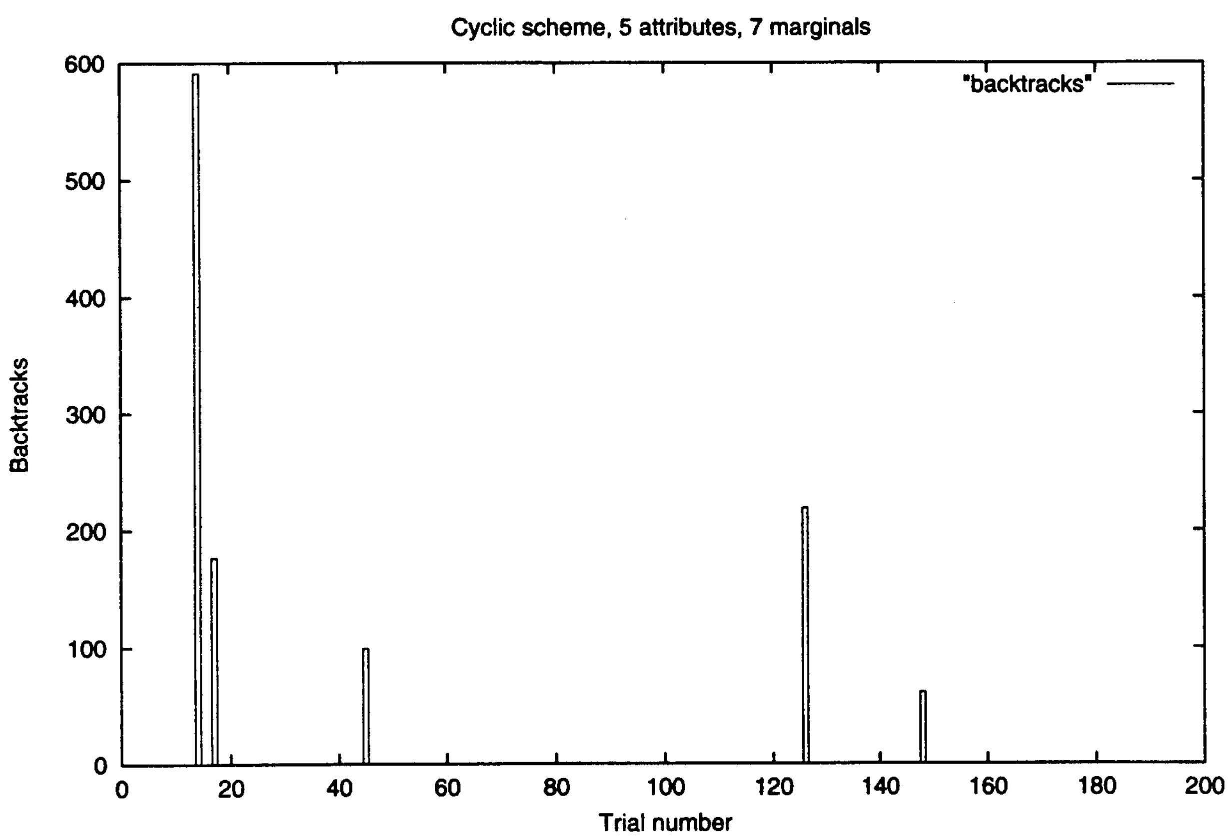
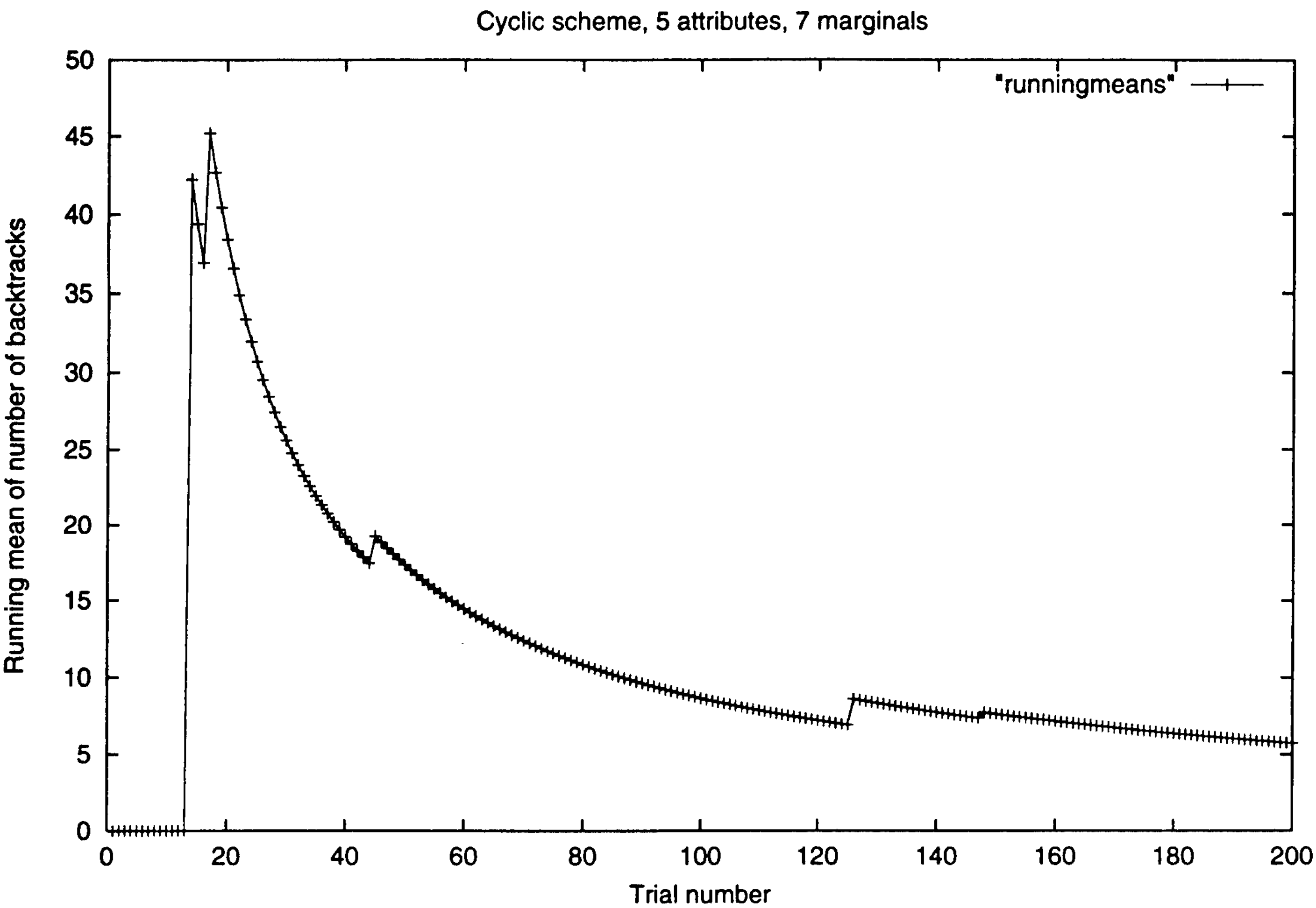


Figure 10



Cyclic scheme, 5 attributes, 7 marginals
"runningmeans"
Running mean of number of backtracks
Trial number
0
5
10
15
20
25
30
35
40
45
50
0
20
40
60
80
100
120
140
160
180
200

Figure 11

Database Inconsistency Problem

A1	A2	f1
0	0	
0	1	12
1	0	
1	1	

A2	A3	f2
0	0	0
0	1	0
1	0	
1	1	16

A1	A3	f3
0	0	0
0	1	
1	0	
1	1	

A1	0	0	0	0	1	1	1	1
A2	0	0	1	1	0	0	1	1
A3	0	1	0	1	0	1	0	1
f4	0	0	0	0	0	0	0	0

Preplace Marginals | Complete Database | Clear Frequencies

Number to Preplace = 5 Sum of Frequencies = 16

STATUS: Marginal frequencies preplaced

Figure 12

ERRORS INHERENT IN THREE-DIMENSIONAL TARGET RECONSTRUCTION FROM MULTIPLE AIRBORNE IMAGES

Mark S. Schmalz
Assistant Scientist
Department of Computer and Information Science and Engineering

University of Florida
Bldg. CSE/E301
Gainesville, FL 32611

Final Report for:
AFOSR Summer Research Extension Program
AFRL/IF, Rome NY

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

POC: Mr. Fred Rahrig
IFEC, Air Force Research Laboratory
Rome, NY

29 December 1998

ERRORS INHERENT IN THREE-DIMENSIONAL TARGET RECONSTRUCTION FROM MULTIPLE AIRBORNE IMAGES

Mark S. Schmalz
Assistant Scientist
Department of Computer and Information Science and Engineering
University of Florida

Abstract

Automated target recognition has benefited from cross-fertilization of development in related subdisciplines of image processing such as medical imaging. For example, the application of computerized tomography to synthetic aperture radar (SAR) imaging has produced 3-D reconstructions of ground targets on an experimental basis. In practice, by acquiring multiple views of a target (also called *multi-look imaging -- MLI*) that are subsequently merged mathematically, one can obtain reasonable approximations to higher-dimensional reconstructions of a target of interest. For example, multiple two-dimensional airborne images of ground objects can be merged via the Fourier transform (FT) to obtain one or more approximate three-dimensional object reconstructions. Additional methods of 3D model construction (e.g., from affine structure) present advantages of computational efficiency, but are sensitive to positioning errors.

In this study, an analysis of MLI is presented that applies to various scenarios of nadir, near-nadir, or off-nadir viewing with a small or large number of narrow- or wide-angle views. A model of imaging through cover describes the visibility of a given target under various viewing conditions. The model can be perturbed to obtain theoretical and simulated predictions of target reconstruction error due to (a) geometric projection error, (b) focal-plane quantization error and camera noise, (c) possible sensor platform errors, and (d) coverage of looks. An information-theoretic model is derived from the imaging model that can facilitate prediction of limiting sensor geometry and view redundancy under various imaging constraints (e.g., target and cover geometry, available range of look angles, etc.). Mathematical models are implemented in a software simulation model that is coded in C and runs on UNIX platforms. Additional research emphasizes analysis of several target reconstruction algorithms based on estimated errors in MLI data and systematic error sources such as computational error in arithmetic operations. Implementational discussion concerns the employment of the computer simulation model developed under the scope of this study in applications such as location of submerged targets from airborne imagery, and prediction of optimum sensor track given partial knowledge of target position and size as well as cover geometry and location.

Study notation is a subset of image algebra, a rigorous, concise, computationally complete notation that unifies linear and nonlinear mathematics in the image domain [Rit96]. Image algebra was developed at University of Florida over the past decade under the sponsorship of DARPA and the US Air Force, and has been implemented on numerous sequential workstations and parallel processors. Hence, our algorithms are rigorous and widely portable.

TABLE OF CONTENTS

ERRORS INHERENT IN RECONSTRUCTION OF TARGETS FROM MULTI-LOOK IMAGERY

Mark S. Schmalz

1. Introduction

Automated recognition of ground targets from airborne imagery can be confounded by vegetative or manmade cover that overlies targets of interest. For example, consider a vehicle parked beneath an overhanging tree line or concealed under camouflage netting. Under such circumstances, it would be advantageous to acquire and exploit multiple target images at different look angles (called *multi-look imaging* or *MLI*) [Lon81,Chr96,Moo96,Sha96,PAR97]. Selected MLI images could be combined mathematically to yield a composite view of a target of interest. With appropriate supporting models, this process is being implemented in computer software that produces simulated MLI imagery and error analyses [Sch97a,b].

In this study, the following objectives are accomplished:

1) Understand and model the phenomenology of multi-look imaging;
2) Develop models and error analyses for tomographic or tomography-like target reconstruction from multiple airborne views of ground targets;
3) Apply theory and models to the development of computer software that estimates errors in salient system parameters under given sensing geometry constraints; and
4) Acquire realistic laboratory or field imagery to facilitate algorithm design and test.

Computer model(s) are summarized in Section 4.

1.1. Document Organization

This report begins with a review of previous work (Section 1.2) and a summary of key study issues and assumptions (Section 2), followed by a presentation of theory and methodology that describes basic MLI phenomenology (Section 3). Theory is extended to yield an information-theoretic model of target visibility in MLI scenarios, which supports theory development for tomographic reconstruction. The result of this study, a preliminary analysis of the difficult and error-prone process of target reconstruction from multiple views is discussed in Section 4. Conclusions and suggestions for future work are presented in Section 5.

1.2. Previous Work

Detection and recognition of manufactured targets in natural scenes is based on the selection of a model that best matches a subregion of an observed image. When an imaging sensor's view of a target is unobstructed, the matching process can be implemented via 1) least-squares minimization in image space, or 2) suboptimal methods using correspondence subsets. In the presence of partial occlusion, one could 3) progressively reconstruct the object via coregistration and merging of multiple views or 4) employ tomography-like methods that approximate a solid model of the target. The tomographic model could be serially sectioned with respect to a reference vector, thereby yielding a layered model of objects located between ground and cover. Methods 1) through 3) are summarized in Sections 1.2.1-1.2.3, followed by comparison of three-dimensional (3-D) models and 2-D images in Section 1.2.4. We do not consider the limiting and extremely difficult case of total occlusion, which is outside the scope of this study.

1.2.1. Least-squares minimization techniques. Traditional photometric approaches to matching images and reference models involve least-squares minimization of the Euclidean distance between the model and an elevation map constructed from the image. This problem is generally classified as an *exterior orientation calibration problem* (EOCP) whose solution, given a set of 3-D (2-D) model (image) points **U** (**V**), finds a rigid transformation that minimizes a distance between **U** and **V** [Bas96]. Since analytical solutions to the EOCP remain undiscovered, numerical methods are employed, which tend to be unstable in the presence of discontinuities and noise, are computationally costly, and require accurate initial values.

Marr and Nishihara [Mar78] first addressed the numerical problems of describing 3-D structure systematically. Minsky [Min75] had previously suggested that the complexity of shape description could be minimized by appropriate choice of primitives, although this has yet to be realized in general theory. In the intervening two decades, surface reconstruction from a single view has been implemented in terms of variational methods [Mei79,Sch77,Ter83,Ter88], segmentation and synthesis or a variety of ad hoc techniques [Bes85]. Bolle and Vemuri [Bol91] summarize variational surface fitting techniques that (a) have varying degrees of viewpoint invariance as well as robustness in the presence of noise, (b) tolerate bias in parameter estimates, and (c) exhibit various sensitivities to obscuration. The following discussion is derived from the excellent overviews given in [Bol91] and [Vem86,87].

In its simplest form [Bar81], surface fitting involves interpolation of uniformly curved surfaces from initial orientation values and constraints using a relaxation algorithm based on parallel iterative local averaging. Unfortunately, this technique ignores surface discontinuities, is computationally costly, and produces a surface representation that is invariant to 3-D rigid motion.

Grimson [Gri81] presented a theory of visual surface interpolation based on range data from stereo imagery that were obtained via Marr and Poggio's correspondence algorithm [Mar79]. Given a finite image domain $\mathbf{X} \subset \mathbf{R}^n$ and an elevation map $\mathbf{a} \in \mathbf{R}^{\mathbf{X}} \equiv \mathbf{X} \to \mathbf{R}$, Grimson minimized the variation of the quadratic functional

$$E = \int_N \mathbf{a}_{xx}^2 + 2\mathbf{a}_{xy}^2 + \mathbf{a}_{yy}^2 \; dx\, dy \tag{I}$$

where $(x, y) \in \mathbf{X}$, $N \subset \mathbf{X}$, and $\mathbf{a}_{xy} \equiv \partial x / \partial y$. E denotes the energy of a thin plate, hence its minimization yields a function called *thin plate splines* [Mei79]. Grimson developed an iterative algorithm based on biharmonic convergence that results from applying Euler's equations to minimize E. Although invariant to rotation and translation, Equation (I) is not invariant to viewpoint, since E depends on the coordinate system in which depth constraints are specified. Additionally, Grimson's model did not admit discontinuities.

In contrast, Terzopoulos [Ter83,88] derived an efficient surface model that accepts multiresolution imagery, is computationally efficient, and accounts for depth and orientation discontinuities. Surfaces are modeled as thin plate segments joined by membrane strips along orientation discontinuities and bounded by depth discontinuities. The model is invariant to translation and rotation but not to rigid motion, and is based upon a continuity stabilizer described in [Ter83]. Harris' extension [Har86] of Terzopoulos' model couples depth and slope, this integrating orientation constraints into the minimization of the energy functional E. Harris' model admits varying surface smoothness, parallel implementation, discontinuities of any order, and generalizes splines under tension since it can integrate arbitrary combinations of membrane, thin-plate, or high-order smoothness constraints.

In MLI imagery, the surface fitting method would be complicated by the lack of registration between views that could exhibit geometric distortion, rotation, and scale differences. Hence, a parameterized method (e.g., quadric or superquadric representation) would be required that could be scaled, rotated, and translated to implement coregistration between views in model space (versus customary image-domain coregistration). Sensor noise, projection error, and quantization error would thus increase surface fitting error. More importantly, the greyscale variations induced by patterns of light and shadow or camouflage devices could easily confound the construction of an elevation map based on well-known shape-from-shading techniques, as discussed in [Sch97b].

1.2.2. Suboptimal matching with correspondence subsets. The change in the 2-D projection of a moving 3-D object or camera yields important information for 3-D object reconstruction that elucidates relative geometry. The problem of computing shape from motion or a sequence of images has been addressed using calibrated or uncalibrated cameras [Fau92] as well as projective or affine reconstruction models [Fau95]. Camera calibration facilitates the computation of Euclidean shape up to a scale factor using projective or affine models [Der94]. In the absence of calibration, the recovered shape is defined up to a projective or affine transformation. Although weaker than that provided by surface fitting from depth or elevation maps, or by inversion of projection equations with fully specified camera calibration matrices, relative geometry is useful since

1. Reconstruction algorithms can be made simpler and more easily parallelized.
2. The costly and time-consuming process of camera calibration is not required;
3. Multiple views of an object need not be equally spaced to obtain regularity; and
4. Distinction between orthographic and perspective projections is not required.

In practice, one (a) chooses a representation of projective space in which an arbitrary reference plane is assumed to be located at infinity, (b) describes the new representation by an element of an affine group applied to the initial representation, and thus (c) obtains an affine invariant relative to the initial representation.

Affine reconstruction is expected to be a useful target reconstruction technique in conjunction with MLI. Three recent papers highlight associated technical issues:

- *Christy & Horaud* [Chr96] incrementally perform Euclidean reconstruction based on a weak-perspective or paraperspective (calibrated) camera model. Fast convergence (in a few iterations), computational efficiency, and solution of the sign reversal ambiguity problem are cited as advantages.
- *Moons et al.* [Moo96] describe affine reconstruction of a scene from two views with relative target-camera translation between views. The reported approach requires only five corresponding points, which can be obtained by well-known coregistration algorithms [Yes89,Smi96], unless significant anisomorphic distortion is present in coregistered views. This is usually the case for multi-look imagery where a majority of the views are taken over a narrow angular range.
- *Shashua & Navab* [Sha96] describe an affine reconstruction technique for perspective views that unifies Euclidean, projective, and affine models. Simple algorithms are presented for reconstruction from multiple views, recognition by alignment of segmented objects with templates, and coding of video sequences.

We see affine reconstruction as a possible alternative to tomographic reconstruction that is attractive implementationally due to viewpoint invariance [Sha96], lack of camera calibration requirements [Sha96, Moo96], and tolerance of camera translation [Moo96]. More detailed discussion is presented in [Sch97b].

Shashua and Navab analyzed interpolation and extrapolation errors. Given three views $\mathbf{a}_{i-1}$, $\mathbf{a}_i$, and $\mathbf{a}_{i+1}$, extrapolation based on the (i-1)-th and i-th views was employed in reconstructing an approximation to the (i+1)-th view. Error was measured as the mean squared distance between corresponding points in the extrapolated and actual views. As expected, the upper bound on error was provided by extrapolation, whose error decreased to 1.1 ± 0.98 pixels as the separation between the i-th and (I+1)-th views increased, up to the visibility limit. This is due to the fact that extrapolation is usually more erroneous than interpolation, given identical continuity assumptions. Scene reconstruction error was measured as the mean-squared difference between predicted and measured object elevation. A cube having greyscale patterns on its faces was employed as a test object. The average depth error was 0.23 ± 0.31 percent of mean range distance r to target, with a peak-to-peak error range of 0.002r to 0.007r.

Practical considerations in affine-based 3-D modeling for MLI applications emphasize the accuracy with which the various points can be located in terms of world or relative coordinates. In imagery, pointing accuracy is limited by focal plane quantization error. The accurate determination of sensor position and orientation is crucial to establishing the camera origin points. If the separation between sensor origins can be determined accurately, then Shashua's method [Sha96] can be modified to yield displacement-based measures of relative affine structure, per [Moo96].

1.2.3. Coregistration-based techniques. Another approach to multi-look imaging involves coregistration of the multiple looks across projections (views) and spectral bands (in the case of multispectral imagery). If the spectral bands are not severely decorrelated temporally, current coregistration technology may be adequate for inter-band alignment within ± 1 pixel. However, each projection (taken at a different look angle and altitude) has scaling, rotation, shift, and anisomorphic geometric distortions that tend to make registration across all views difficult for wide-angle MLI.

Yeshurun and Schwartz first proposed cepstral filtering for region-based solution of the stereo correspondence problem [Yes89], which instantiates a correspondence problem. This scheme uses an adaptive prefilter followed by autocorrelation, which provi i :s prewhitening that reduces the noise sensitivity of the resulting stereo disparity. Given a stereo pair of MxN-pixel images, the stereo image formed as **c** = (**a**,**b**) is transformed via application of the cepstrum to yield the disparity image **d**, as follows:

$$\mathbf{d} = \mathcal{F}(\log[|\mathcal{F}(\mathbf{a}) + 1|]) \quad ,$$

where $\mathcal{F}$ denotes the Fourier transformation. The difference between the coordinates of the origin and largest peak of **d** yields the stereo disparity. Since the cepstral approach can be applied to small neighborhoods of an image that contain corresponding points, it is useful in principle for imagery that has space-variant distortion. Unfortunately, the location of correspondence neighborhoods is itself an instance of the correspondence problem that may simplify the problem implementationally via a hierarchical registration scheme, but does not fundamentally reduce complexity.

Ludwig et al. [Lud94] performed an error analysis of Yeshurun and Schwartz' technique, to determine the performance of cepstral-based correspondence as a function of image noise and distortion. It was suggested that LoG windowing functions be employed for vignetting each region to which the cepstrum is applied. This reduces disparity artifacts slightly without improving algorithm robustness in the presence of scaling distortion or photometric variance.

Smith and Nandhakumar [Smi96] employ the cepstrum as a nonlinear correlator to achieve increased tolerance of noise and anisomorphic geometric distortions. Their enhancements to Yeshurun and Schwartz' technique (a) replace the peak detection process in the power cepstrum with a more robust search mechanism in disparity space and (b) correct for foreshortening effects. The result is a reduction in sensitivity of the corresponding disparity measure to photometric variation and anisomorphic distortions.

Given the foregoing techniques, advantages of coregistering MLI images include:

- Effects of random sensor noise can be ameliorated somewhat by accumulation of the coregistered images, and
- One may be able to obtain depth information for common points across all views of an MLI image sequence, for purposes of 3-D model construction.

Unfortunately, the following practical disadvantages can accrue from erroneous statistical assumptions, computational error, and continuity assumptions required for coregistering multiple views:

- Lateral spatial alignment (coregistration) error may induce elevation errors in stereo reconstruction of depth maps from MLI sequence pairs, which errors may exceed salient target feature height; and
- Computational error inherent in the resampling process required to rectify (coregister) imagery to compensate disparity effects may also induce elevation errors in the reconstructed 3-D target model.

Although region-based cepstral correspondence matching is currently our method of choice for image coregistration, additional research remains in global coregistration techniques. Further discussion and error analyses are given in [Smi96] and [Yes89].

1.2.4. Comparison of 3-D models and 2-D images. Given an image **a**, an object recognition process typically extracts key features, then seeks a model M that best matches the closest view of the feature ensemble [Fis81,Hut90,Ull91]. Image-model alignment can be inexact due (for example) to input noise, model specification errors, and deformation in nonrigid objects. Thus, robust assessment of alignment accuracy is required. The common assumption of a Euclidean distance (called the *image metric* [Bas96]) between image feature points and their corresponding points in the model view of an object that best matches imagery assumes that images (being inherently more erroneous than models) induce alignment perturbations that should be measured in the image plane. Although this assumption may be suitable for recognition, object classification tends to benefit more from minimal object deformation that tolerates uncertainties in a test object's structure.

Measures that compare 3-D models and 2-D images should be *metrical*, which Basri [Bas96] defines as increasing monotonically with the difference between M and **a**. To achieve this, the class of transformations applied to the object are customarily extended from rigid to affine transforms. Although this bounds the rigid measure from below, an upper bound is not attained. Previous methods tended to achieve suboptimal distances or did not produce a metric.

Basri and Weinshall [Bas96] reported a distance measure f that compares a 3-D model M with a 2-D image **a** in closed form. This penalizes nonrigidity incurred by an optimal affine transformation T that aligns M with **a** under weak-perspective projection, as follows:

$$f(M, \mathbf{a}) = \bigwedge_{R \in \mathrm{T}_{R_g}} \|T(M, \mathbf{a}) - R(M, \mathbf{a})\|^2 ,$$

where T_{Rg} denotes the set of all rigid transformations and the norm is denoted by (|| ||). Thus f, which bounds the least-squares distance from above and below, facilitates

1. Direct assessment of similarity between 3-D models and 2-D images;
2. Obtaining upper and lower bounds on the *image metric* for object classification; and
3. Derivation of an initial value set for numerical surface fitting procedures.

An additional application is the evaluation of hypothesized correspondences in *alignment algorithms*, which evaluate the similarity between models and images based on sparse point-to-point correspondences. Although this technique can produce polynomial-time matching in the presence of occlusion [Gri92], matching errors invariably result due to sparse sampling, which can be mitigated with additional correspondences [Fis81]. The measure f can be further employed to evaluate the amount of affine distortion applied to an object relative to a prototype, thereby determining the best-match object class or, in certain cases, object identity.

2. Key Issues and Assumptions

We assume that there exist the following three primary scenarios for MLI:

1. *Narrow-angle, Few Looks* -- the sensor look angle is perturbed only slightly from a reference orientation (customarily nadir viewing) and a few (e.g., 5 to 10) images are taken over a relatively narrow field-of-view (e.g., 10 to 20 degrees).

2. *Wide-angle, On-track* -- the sensor moves along an approximately linear or curvilinear track, and acquires imagery over few or many looks, over a wide field of view.
3. *Loitering* -- the target is reconnoitered by an airborne sensor describing a simple or complex curvilinear path, with unequally spaced views from different elevations.

Scenario 1 has the potential of producing approximately equally spaced looks, some of which can be corrupted by cover interposed between sensor and target. This is the scheme proposed in [PAR97], upon which tomographic reconstruction of the target would be based. Scenarios 2 and 3 would typically acquire unevenly spaced images, with scale differences likely as sensor altitude (and, possibly, camera magnification) changes unpredictably. We have found that such schemes are often amenable to 3D target reconstruction via affine structure extraction or surface fitting. Tomography-based approaches would be confounded by large partitions of missing information in the tomographic reconstruction space, especially if few looks were taken.

3. Methodology

We begin with a one-dimensional model, for purposes of illustration, then progress to a two-dimensional model of cover, which is instantiated in computer software.

3.1. Theory of Multi-Look Imaging

Assumption 1. Let a spatial domain $\mathbf{X} \subset \mathbf{R}$ represent (for example) a background region viewed by an imaging sensor at altitude a_s, as shown schematically in Figure 1. Let opaque cover located at altitude a_c be comprised of segments having width w_i, where $1 \le i \le n$. The segments are separated by horizontal apertures g_i that have centroids at $x_i^c \in \mathbf{X}$. Per Figure 1, we assume that the i-th aperture precedes the i-th cover segment as one proceeds away from the sensor.

Lemma 1. *Given Assumption 1, if the sensor images a contiguous partition* $[x_i^{min}, x_i^{max}]$ *of* $\mathbf{X}$ *through aperture , then the following statements hold:*

1. *Look angle extrema* $\phi_v^{min}(i) = \tan^{-1}\left(\frac{\sum_{j=1}^{i-1} g_j + w_j}{a_s - a_c}\right)$ *and* $\phi_v^{max}(i) = \tan^{-1}\left(\frac{g_i + \sum_{j=1}^{i-1} g_j + w_j}{a_s - a_c}\right)$;

2. *Partition extrema* $x_i^{min} = a_s \cdot \tan(\phi_v^{min})$ *and* $x_i^{max} = a_s \cdot \tan(\phi_v^{max})$; *and*

3. *Field of view* $v_i = x_i^{max} - x_i^{min}$.

Proof. The proof follows directly from the geometry of Figure 1. ■

The preceding model does not suffer loss of generality, since if all indices i are in the interval $[1,n]$, then $w_i = 0$ depicts the case of no cover and describes the situation where there are no apertures. Alternatively, if a given aperture subtends **X**, then there is no cover but viewing is unobstructed. In practice, the model of Figure 1 is best applied when a target is situated against the background and the cover segments w_i are vertically thin in relation to the sensor or cover altitude. We also assume that the apertures g_i are sufficiently large to render diffraction effects negligible.

Additionally, the preceding one-dimensional model can be generalized to higher-dimensional domains by observing that, in multi-look imaging, the received image **a** is comprised of more than one projection of the target restricted to the corresponding projection of the cover. Since the cover is assumed to be opaque, application of restriction in this context does not compromise rigor. Hence, the one-dimensional model and its generalization in two or more dimensions are sufficiently general and rigorous to portray physical reality in this preliminary study.

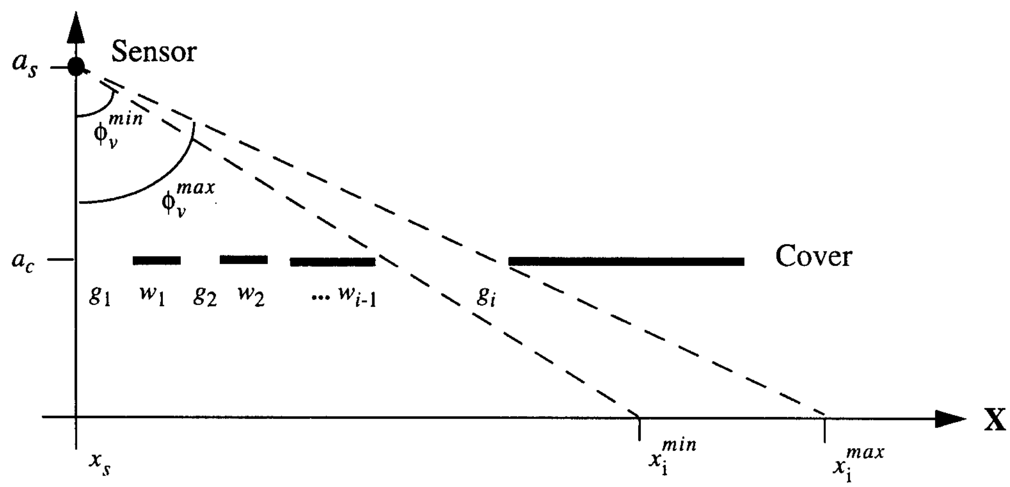


Figure 1. Schematic sensor geometry for multi-look imaging through perforated cover, where x_s denotes lateral sensor position.

Assumption 2. Let domain $\mathbf{X} \subset \mathbf{R}^2$ and assume that cover is specified by an image $\mathbf{c} \in \mathbf{R}^{\mathbf{X}}$ located a_c meters (m) above a target whose elevation map (or equivalent projection) is denoted by the Boolean image $\mathbf{b} \in \mathbf{B}^{\mathbf{X}}$. In this study, we adopt the convention that the values of **c** denote the transmission coefficients of the cover. Let an imaging sensor of focal length f and magnification M be located a_s meters above the target plane, as shown in Figure 2. For each configuration of cover $\mathbf{c}_i$ and target $\mathbf{b}_i$, where $1 \le i \le n$, let the sensor form an image $\mathbf{a}_i \in \mathbf{F}^{\mathbf{X}}$. Further assume that the transmissive medium between target and cover (cover and sensor) has absorption and scattering coefficients a_{tc}, b_{tc} (a_{cs}, b_{cs}) and volume scattering function β_{tc} (β_{cs}).

Observation. When simulating the formation of sensor image **a**, one of two cases pertain:

Case 1: A flat target (or one that can be assumed to be flat), where **b** denotes a reflectance map, or

Case 2: 3-D target, where **b** denotes an elevation map.

For purposes of illustration, the following development is specific to the first case, and the second case is discussed beginning with Theorem 3.

Definition 1. A *spatial transformation* $f : \mathbf{Y} \rightarrow \mathbf{X}$ is a map between spatial domains. Given an image $\mathbf{a} \in \mathbf{F}^{\mathbf{X}}$, a spatially transformed image $\mathbf{b} \in \mathbf{F}^{\mathbf{Y}}$ is denoted by

$$\mathbf{b} = \mathbf{a} \circ f \equiv \{(\mathbf{y}, \mathbf{a}(\mathbf{y})) : \mathbf{y} \in \mathbf{Y}\} .$$

Theorem 1. *Given Assumption 1 and the geometry of Figure 2, the projection S of a point* $\mathbf{x} \in domain(\mathbf{c})$ *to a point* $\mathbf{y} \in domain(\mathbf{d})$ *is given by*

$$\mathbf{x} = S(\mathbf{y}) = \left(\frac{a_s - a_c}{a_s}\right) \cdot h' \cdot (\sin\theta_v, \cos\theta_v) ,$$

where $h' = ([p_1(\mathbf{y})]^2 + [p_2(\mathbf{y})]^2)^{1/2}$ *and* $\theta_v = \tan^{-1}(p_2(\mathbf{y})/p_1(\mathbf{y}))$, *with* $p_k(\mathbf{y})$ *denoting projection to the k-th coordinate, which implies that* $\mathbf{d} = \mathbf{c} \circ S$.

Proof. From the similar triangles of Figures 1 and 2, the target dimension along the x- and y -axis of *domain*(**b**) is scaled by $(a_s - a_c)/a_s$ in *domain*(**c**). This scale factor is also applied to the hypotenuse h' to yield h in *domain*(**c**). The hypotenuse h' is projected via angle θ_v which does not vary between **b** and **c**. Hence, we have the expression $S(\mathbf{y}) = ((a_s - a_c)/a_s) \cdot h' \cdot (\sin\theta_v, \cos\theta_v)$, and $\mathbf{d} = \mathbf{c} \circ S$ follows from Definition 1. ∎

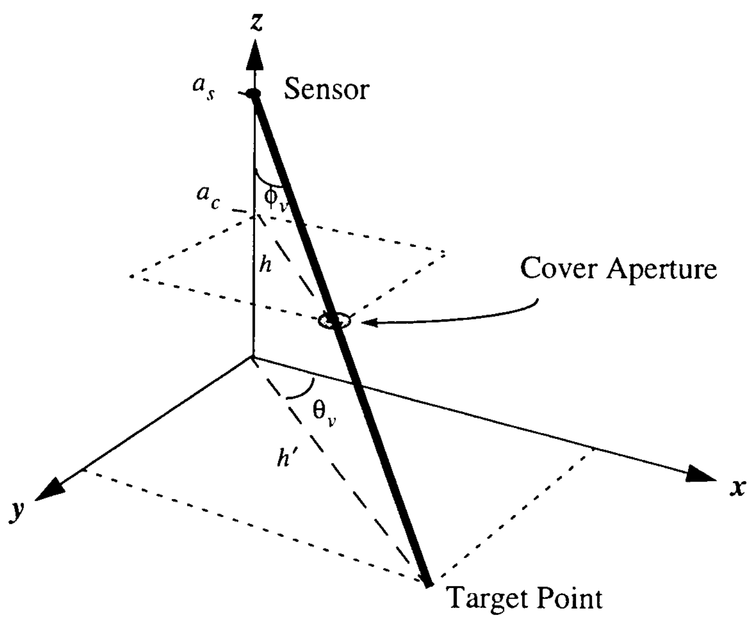


Figure 2. Three-dimensional sensing geometry.

Theorem 2. *Given a volume scattering function* $\beta : (-\pi, \pi] \rightarrow [0, 1]$ *and a scattering distance* $d \in \mathbf{R}^+$ *then a template* $\mathbf{s} : \mathbf{R}^+ \times \mathbf{X} \rightarrow \mathbf{R}^{\mathbf{X}}$ *that instantiates a single-scattering, near-field approximation of* $\beta(\vartheta)$ *is defined in terms of its weights as*

$$\mathbf{s_y}(d)(\mathbf{x}) = \beta(\vartheta)\,,\ where\ \vartheta = \tan^{-1}\left(\frac{\|\mathbf{y}-\mathbf{x}\|}{d}\right).$$

Proof. Given the geometry of Figure 3, it suffices to state that $\vartheta = \tan^{-1}(\|\mathbf{y}-\mathbf{x}\|/d)$. ■

3.2. MLI Model

We next present several algorithms and theorems that we are currently instantiating in a computer model, which would simulate the effect of multiple views of a target object through perforated cover.

Algorithm 1. Given Assumption 1, if $\mathbf{b}$ denotes a greyscale image (e.g., a reflectance map), then the formation of image $\mathbf{a}_i$ can be simulated by performing the following steps:

Step 1. Form the spatial transform $S_{ii} : \mathbf{X} \to \mathbf{X}$, from which the projection of $\mathbf{c}_i$ to the target plane is given by $\mathbf{c} \circ S$, per Definition 1. From S, sensor geometry, and the geometry of $\mathbf{X}$, one obtains a map $\mathbf{d} \in (\mathbf{R}^+)^{\mathbf{X}}$ of the propagation distance from the target plane to the sensor as a function of a point $\mathbf{x} \in \mathbf{X}$, as shown in Theorem 2.

Step 2. Given the media optical parameters a_{tc}, b_{tc}, and β_{tc}, form the attenuation template $\mathbf{s}_{tc} : \mathbf{R}^+ \times \mathbf{X} \to \mathbf{R}^{\mathbf{X}}$, which is parameterized by the propagation distance between target and cover, per Theorem 2. An additional attenuation template $\mathbf{s}_{cs}$ is formed symmetrically from a_{cs}, b_{cs}, and β_{cs} that portrays attenuation in the propagation path segment from cover to sensor.

Step 3. Apply S, $\mathbf{s}_{tc}$, and $\mathbf{s}_{cs}$ to $\mathbf{c}$ to yield $\mathbf{s}_i = \mathbf{s}_{tc}(\mathbf{d}_i) \oplus \mathbf{s}_{cs}(\mathbf{d}_i)$ and an approximation to $\mathbf{a}_i$, as follows:

$$\mathbf{a}_i = [\mathbf{b} * (\mathbf{c}_i \circ S_i)] \oplus \mathbf{s}_i\,, \qquad \text{(II)}$$

The preceding equation does not include realistic receiver effects, such as noise, nonuniform pixel gain, or missing pixels, which we have discussed in [Sch96a], and which would be included in the completed model.

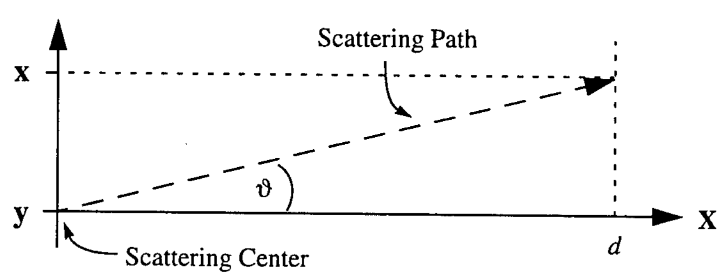


Figure 3. Linear approximation for single-scattering geometry.

Theorem 3. *Let a cylindrical target of radius r and height h be viewed by an ideal camera of unitary magnification, at look angle ϕ measured away from the vertical). The apparent target dimension is given by* $h \cdot \cos\phi + r \cdot \sin\phi$.

Proof. The proof follows directly from the construction of Figure 4a. ■

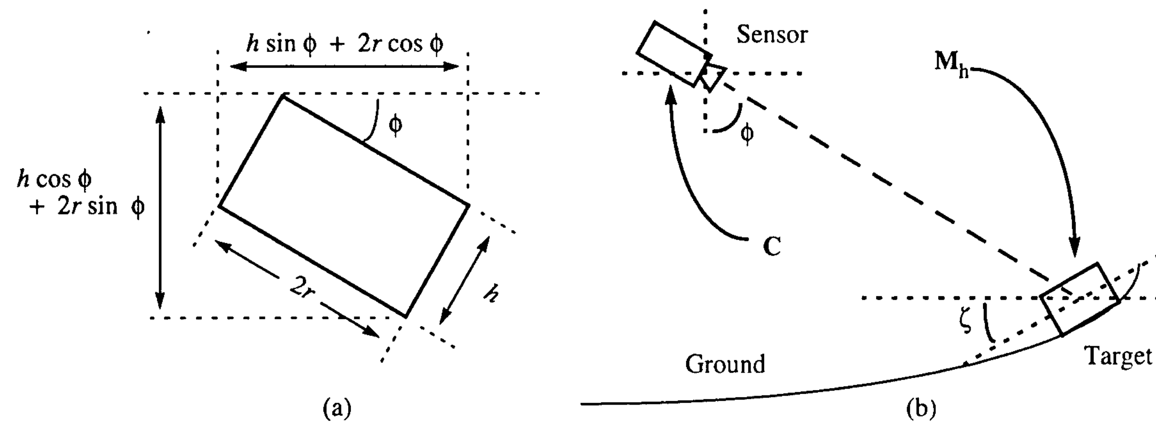


Figure 4. Three-dimensional target (a) geometry, and (b) sensing configuration.

Theorem 4. *Given the geometry of Figure 4b, if* **P** *denotes a perspective correction matrix;* **C**, **R**, *and* **G** *denote camera, rotation, and translation matrices, respectively; and* $\mathbf{M}_h$ *denotes local target coordinates; then the target coordinates projected to the focal plane of a sensor at altitude* a_s *are given by* $\mathbf{C} = (x, y, z, 1)^T$, *such that* $\mathbf{C}_f = \mathbf{P\,C\,R\,G\,M}_h$.

Proof. Let $\mathbf{M}_h$ denote local target coordinates, and let translation matrices **G** and **C** be given by

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & -p_1(\mathbf{p}) \\ 0 & 1 & 0 & -p_2(\mathbf{p}) \\ 0 & 0 & 1 & -p_3(\mathbf{p}) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & -p_1(\mathbf{p}) \\ 0 & 1 & 0 & -p_2(\mathbf{p}) \\ 0 & 0 & 1 & -p_3(\mathbf{p}) \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where **r** denotes the coordinates of the sensor camera's focal point, with the camera lens nodal point located at **p**. Let $\mathbf{R} = \mathbf{R}_\phi \mathbf{R}_\theta$ be a combination of two rotations $\mathbf{R}_\phi$ (tilt angle) and $\mathbf{R}_\theta$ (pan angle), such that

$$\mathbf{R} = \begin{pmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta\cos\phi & \cos\theta\cos\phi & \sin\phi & 0 \\ \sin\theta\sin\phi & -\cos\theta\sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Define the perspective projection matrix **P** as follows:

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/f & 1 \end{pmatrix},$$

where f denotes the sensor focal length. Assume that translation and rotation matrices **T** and **R** are given by

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 & -x_m \\ 0 & 1 & 0 & -y_m \\ 0 & 0 & 1 & -z_m \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{T}^{-1} = \begin{pmatrix} 1 & 0 & 0 & x_m \\ 0 & 1 & 0 & y_m \\ 0 & 0 & 1 & z_m \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \text{and} \quad \mathbf{R} = \begin{pmatrix} 1 & 0 & 0 & -x_m \\ 0 & \cos\zeta & \sin\zeta & -y_m \\ 0 & -\sin\zeta & \cos\zeta & -z_m \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

From the expression of Theorem 3 in two dimensions, and letting r and h denote the radius of height of a cylindrical target inclined at an angle ζ from the horizontal, we have that $\mathbf{M}_h = (h\sin\phi + 2r\cos\phi,\ h\cos\phi + 2r\sin\phi,\ h_0)$, where $\phi \in [0, 2\pi)$ and $|h_0| \le h/2$. One thus obtains

$$\mathbf{W}_h = \mathbf{T}^{-1}\mathbf{R}\ \mathbf{T}\ \mathbf{M}_h ,$$

which yields the projection of target coordinates to image space as

$$\mathbf{C} = \mathbf{P}\,\mathbf{C}\,\mathbf{R}\,\mathbf{G}\,\mathbf{M}_h . \qquad \blacksquare$$

3.3. Theory of Tomographic Reconstruction

Computerized axial tomography customarily projects a planar beam of X-rays through an object, where the transmitted X-rays are measured by a linear array of detectors. By rotating the transmitter/detector through 180 degrees at small, fixed angular increments then applying a reconstruction algorithm, a high-resolution of a given cross-section through an object may be obtained. Combining these cross-sections into a solid model yields 3-D information concerning the object of regard. In this study, we concentrate on airborne MLI imagery of battlefield scenes, and investigate whether or not tomographic imaging techniques could be applied to such imagery. We call this process *battlefield tomography*, and note that control of the sensing parameters or knowledge of target/media properties would not necessarily be precise. In contrast, medical tomography has the advantages of (a) well-known optical parameters of various tissues that vary with relative predictability cross-population and among various pathological conditions, (b) a highly constrained geometry where transmitter and detector positions are known accurately, (c) quantization bin-size that can be set sufficiently small to meet diagnostic accuracy constraints, and (d) offline processing capability that admits reasonable computational cost to achieve high reconstruction accuracy.

Assume that a source and detector assembly is used to sense transmittance at each linear increment along the common z axis of superimposed 3-D Cartesian and polar coordinate spaces. Given a two-dimensional domain $\mathbf{X}$, one thus obtains a collection of images $\mathbf{a} \equiv \{(i, \mathbf{a}_i) : \mathbf{a}_i \in \mathbf{R}^{\mathbf{X}}\}$. Given incident beam intensity I_0, $\mathbf{a}$ can be transformed to obtain an in-plane density function

$$\mathbf{d}(\rho, \theta) = \frac{I_0}{\mathbf{a}(\rho, \theta)}, \quad \text{where} \quad \rho = \sqrt{x^2 + y^2} \quad \text{and} \quad \theta = \tan^{-1}(y/x) \quad \text{with} \quad (x, y) \in \mathbf{X},$$

and the beam direction forms angle θ with the y-axis. We assume for purposes of convenience that $\mathbf{d}$ has the same set-theoretic form as $\mathbf{a}$ and can be operated upon by the Radon transform to yield the following expression for the projection process:

$$\mathbf{b}(\rho, \theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathbf{d}(\rho, \theta) \cdot \delta(x \cos \theta + y \sin \theta)\, dx\, dy, (x, y) \in \mathbf{X} ,$$

where δ denotes the Dirac delta function with ρ and θ defined previously.

In the case of CAT scanning, image reconstruction is based upon the similarity properties of the two-dimensional Fourier transform $\mathcal{F} : \mathbf{R}^{\mathbf{X}} \rightarrow \mathbf{R}^{\mathbf{X}}$ Given $\mathbf{c} = \mathcal{F}(\mathbf{d})$ we obtain

$$\mathcal{F}(\mathbf{b}(\rho, \theta)) = \mathbf{c}(\xi, \theta) ,$$

where $\xi = \sqrt{u^2 + v^2}$ and $(u, v) \in domain(\mathcal{F}(\mathbf{b}))$ Thus, each density function yields a function $\mathbf{c}_i$ that represents a radial cross-section through the 2-D FT of the object. By sampling θ in small steps, the resulting transforms can be interpolated to approximate **c**, which can be inverse-transformed to yield a further approximation to **b**. Performed over a range of linear increments along the z-axis, this process yields an approximation to the 3-D X-ray density map of the object.

Assuming (for purposes of brevity) that **c** is defined on domain **X**, the computational efficiency of the FT-based reconstruction algorithm can be increased at the expense of accuracy by employing the *back-projection algorithm*, which is given by

$$\mathbf{c}(x, y) = \int_0^{\pi} \mathbf{b}(x \cos \theta + y \sin \theta,\ \theta)\, d\theta, (x, y) \in \mathbf{X} . \qquad \text{(III)}$$

Each projected density function is projected back (expanded) along the axis of the beam at orientation θ to form a 2-D image that is comprised of 1-D images parallel to the scanning axis. By summing (superposing) all such images over one z-axis bin, an approximation of the cross-sectional view can be obtained at that bin.

In practice, the reconstructed function **c** is comprised of **d** blurred by a point-spread function of form $\mathbf{f}(x, y) = (x^2 + y^2)^{1/2}$. By highpass filtering each 1-D projection in **b** prior to backprojection with a filter whose MTF increases linearly with spatial frequency, one can approximate the inverse Radon transform, from which **d** is obtained. Lowpass filtering of the reconstruction **d** reduces the appearance of random noise at the expense of resolution.

4. Error Analysis and Physical Model Results

Three types of error are expected to predominate in tomography-based reconstruction algorithms. First, occlusion of the target by opaque or translucent cover segments tends to reduce the information about target reflectance (UV and visible wavelengths) and induced or inherent luminosity (UV or infrared wavelengths). This would locally decrease the signal-to-noise ratio (SNR) at a target point visible in a given view of the object. In the presence of point-spread function (PSF) effects, total occlusion at given target points could yield a small amount of useful information due to channel crosstalk, provided that the PSF could be accurately estimated and deconvolved from the received image. However, this is unlikely in battlefield practice without clearly visible reference objects (e.g., manufactured targets such as vehicles) from which one can estimate edge-spread effects [Sch95,Yan95,Sch96b]. In Section 4.1, we discuss this issue in the context of an information-theoretic model of target occlusion.

A second source of error in target reconstruction from multiple views occurs during projection of one or more pixels of a discrete image from the sensor focal plane to a target point. The discrete focal plane has quantization error that is magnified by the projection process. Additionally, media effects such as scattering and attenuation reduce the certainty with which a point may be located in the focal plane. The combined first-order effects of focal plane quantization and PSF effects yield spatial errors that are projected to the domain of the reconstructed target. For example, in near-nadir viewing at low altitudes, such errors tend to be minimized when paraxial approximations are justified by the sensor configuration. In contrast, given wide-field imaging at long range distance (e.g., a cruise missile viewing a distant target on the edge of a clearing, as the missile approaches above treetop level), slant-path effects due to natural or manmade obscurants can severely degrade the received image. If targets are poorly resolved, then quantization error may predominate. The geometric basis for such effects is summarized in Section 4.2 and [Sch97b].

Target reconstruction algorithms present further problems. For example, tomography based on the Fourier transform (per Section 3.3) customarily requires specially designed filtering or data windowing techniques to maximize SNR and minimize artifacts such as high-frequency ringing or vignetting effects [Nat86]. Additionally, the FT has well-known errors that result from division by small-magnitude coefficients, aliasing of high frequency information, etc. The backprojection algorithm given in Equation (III) has associated errors of approximation, as well as computational errors inherent in the transcendental functions, which tend to be dominated by the key error of angular quantization binsize. In fairness, we note that such errors are not absent from geometric reconstruction algorithms, for example, the relative geometry approach of Shashua and Navab [Sha96]. A key problem in geometry-based reconstruction is accurate location of the viewpoint and target reference points. Also, solutions to the correspondence problem are required that have error approaching the Nyquist limit. Errors involved in such processes are discussed extensively in [Nat86, Yes 89 Smi96] and [Sch97b].

Section 4.3 contains a summary of a computer simulation model of MLI processes that was developed during the study and is being enhanced. An additional topic investigated in this study is overviewed in Section 4.4, namely, physical models that can be employed in place of the computer model for fast MLI simulation using detailed target and background environments. The physical modeling methods, which have long been known, suffer from lack of realism at high resolution, but tend to produce useful imagery for early algorithm testing. Furthermore, the model configuration can be easily varied, and results can be verified directly by physical measurement of the sensor parameters.

4.1. Information-Theoretic Model of Target Occlusion

In this section, we generalize the model developed in Section 3 to produce an estimate of information contained in a given projection of an obscured target viewed through perforated cover.

Definition 2. If a process P has n equiprobable outcomes, then the entropy of P is given by $H(\mathrm{P}) = \log(n)$, where the logarithm is taken to the base two.

Definition 3. If a process P has n possible independent outcomes, each of which has probability p_i, then the entropy of P is given by

$$H(\mathrm{P}) = \sum_{i=1}^{n} -p_i \cdot \log(p_i) \ .$$

If the outcomes of P are not statistically independent (which occurs in practice), then the equality relation in the preceding equation is replaced with the inequality ($\leq$).

Assumption 3. Let the probability of viewing through cover (per Figure 1) be expressed in terms of an image $\mathbf{c}$ that is comprised of component probability maps $\mathbf{c}_i \in [0, 1]^{\mathbf{X}}$. Let the component images of $\mathbf{c}$ be coregistered on a common domain $\mathbf{Y} \supseteq \mathbf{X}$, such that the n domain points in the target plane to which a given focal-plane point $\mathbf{x} \in \mathbf{X}$ could project correspond to one and only one point in $\mathbf{Y}$. We adopt the convention that, in the i-th view, if a pixel $\mathbf{c}_i(\mathbf{x})$ is zero-valued, then there is no view through the cover at that location. Otherwise, there is partial (total) information transmission from the underlying image if $0 < \mathbf{c}_i(\mathbf{x}) < 1$ ($\mathbf{c}_i(\mathbf{x}) = 1$). Let us conceptualize MLI imagery as comprised of multiple views $\mathbf{a}_i \in \mathbf{F}^{\mathbf{X}}$, where $1 \leq i \leq n$, which are coregistered on $\mathbf{Y} = domain(\mathbf{c})$ to produce an image $\mathbf{b}$ that is comprised of images $\mathbf{b}_i \in \mathbf{F}^{\mathbf{Y}}$ each of which corresponds to a view in $\mathbf{a}$. Let λ denote a flag value and let the operation $f : [0, 1] \rightarrow (0, 1] \cup \{\lambda\}$ map nonzero values to themselves and zero to λ. Further assume that the operation $h : \mathbf{R} \cup \lambda \rightarrow \mathbf{R}$ is a refinement of the logarithm, such that

$$h(x) = \begin{cases} \log(x) & \text{if } x \in (0, 1] \\ 0 & \text{otherwise} \end{cases} \ .$$

Theorem 5. Given Assumption 3, the entropy of $\mathbf{c}$ is given by

$$H(\mathbf{c}(\mathbf{y})) \leq \sum_{i=1}^{n} -\mathbf{c}_i(\mathbf{y}) \cdot h(f[\mathbf{c}_i(\mathbf{y})]) \ .$$

Proof. The proof follows directly from the givens and Definition 3. ■

Assuming, as before, that $\mathbf{b}$ and $\mathbf{c}$ share a common domain $\mathbf{Y}$, then $H(\mathbf{c}(\mathbf{y}))$ portrays the entropy of a pixel $\mathbf{b}(\mathbf{y}) = (\mathbf{b}_1(\mathbf{y}), \mathbf{b}_2(\mathbf{y}), \ldots, \mathbf{b}_i(\mathbf{y}), \ldots, \mathbf{b}_n(\mathbf{y}))$, where $\mathbf{b}_i(\mathbf{y})$ denotes the i-th coregistered view of the $\mathbf{y}$-th target pixel. Observe that this information is useful for target recognition purposes, as discussed in the following algorithms.

Algorithm 2. Given Theorem 5 and the preceding observation, one could exclude the pixels of $\mathbf{c}$ that have no target information via the operation

$$\mathbf{f} = \chi_{=0}(\mathbf{c}) \ .$$

Algorithm 3. Given Theorem 5 and the preceding observation, it is possible to determine the indices of projections that exceed an entropy thres'1.ıld T, as follows:

$$D = \bigcup_{i=1}^{n} domain(\chi_{>T}(\Sigma\, \mathbf{c}_i)) \ .$$

Thus, one could preselect certain views for likelihood of useful information. This technique also suggests the use of the singular value decomposition (SVD) for determining significant views.

4.2. Projective Errors

From Figure 1, the projection of a point x in a one-dimensional domain **X** to the i-th focal-plane pixel of a sensor's imaging device looking horizontally constitutes the worst case, since the target is in the focal plane periphery. At altitude a_s, focal length f, and magnification M, the projection from a target point $x \in \mathbf{X}$ to the focal plane is given by

$$i = f \cdot \cot\left(M \cdot \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{x}{a_s}\right)\right]\right) \ . \qquad \text{(IV)}$$

Taking the derivative of i with respect to x, we obtain the worst-case focal-plane error ∂i that results from a location error ∂x in the target plane, as follows:

$$\frac{\partial i}{\partial x} = \frac{f\left(1 + \cot\left(\frac{M}{2}\left[2 \cdot \tan^{-1}\left(\frac{x}{a_s}\right) - \pi\right]\right)^2\right)}{a_s^2 + x^2} \ .$$

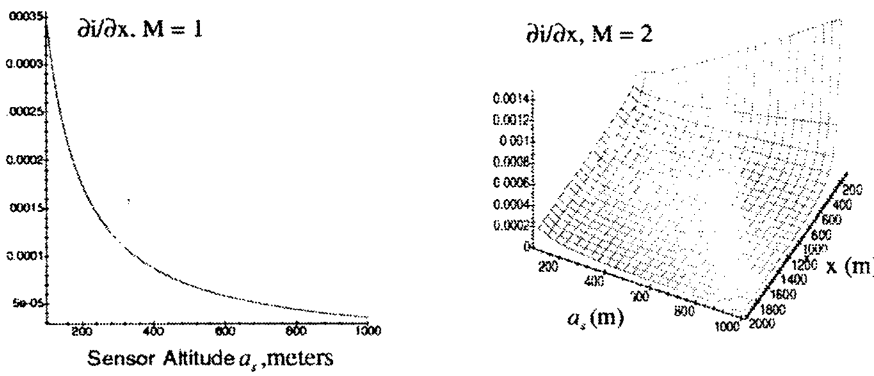


Figure 5. Worst-case target-to-camera projection error (a) M = 1 and (b) M = 2.

Example. Since f and M are usually fixed, we set f = 35mm and M = 1, which are reasonable implementational values. Varying the sensor altitude from 0.1km to 1 km and varying the range distance x from 100m (nadir viewing) to 2km (slant-path viewing), we obtain the result shown in Figure 5a, where $\partial i / \partial x = 0.035 / a_s$. Setting M = 2, we obtain the result of Figure 5b, which is described by the following equation:

$$\left.\frac{\partial i}{\partial x}\right|_{M=2} = 0.07 \cdot \frac{a_s \cdot \left(1 + \cot\left(2 \cdot \tan^{-1}\left(\frac{x}{a_s}\right)\right)^2\right)}{a_s^2 + x^2} \ .$$

The minimum error scenario occurs when the target is on-axis, i.e., when the sensor's optical axis coincides with the line segment between x and a_s. Such error is described by the expression

$$\partial i = f \cdot \tan\left(M\left[\tan^{-1}\left(\frac{x}{a_s}\right) - \tan^{-1}\left(\frac{x + \partial x}{a_s}\right)\right]\right),$$

which reduces to $\partial i \approx \partial(x \cdot f/(Mx))$ in the paraxial assumption.

Example. In the minimum-error case for M = 1, the paraxial approximation at a_s = 400m, x = 1km ground range, and $\partial x = \pm 1\text{m}$ ground error yields a focal-plane error of

$$\partial i \approx \partial(x \cdot f/(Mx)) = \pm 35\mu = \pm 1\text{m} \cdot 1 \cdot 0.035\text{m}/10^3\text{m} .$$

Given 9 micron pixels, $\partial i = \pm 35\mu/9\mu = \pm 3.9$ pixels. With 20 micron pixels, $\partial i = \pm 35\mu/20\mu = \pm 1.7$ pixels.

Under similar constraints, during target recognition, the projection of a ray bundle that subtends the i-th focal plane pixel to a target point $x \in \mathbf{X}$ is given by

$$x = a_s \cdot \tan\left(\frac{\pi}{2} - \frac{1}{M} \cdot \tan^{-1}\left(\frac{i}{f}\right)\right).$$

The general expression for ground error as a function of focal-plane error (the derivative of x with respect to i) is given as follows:

$$\frac{\partial x}{\partial i} = -\frac{a_s\left(1 + \cot\left(\frac{\tan^{-1}(i/f)}{M}\right)\right)}{M(f^2 + i^2)} . \qquad \text{(V)}$$

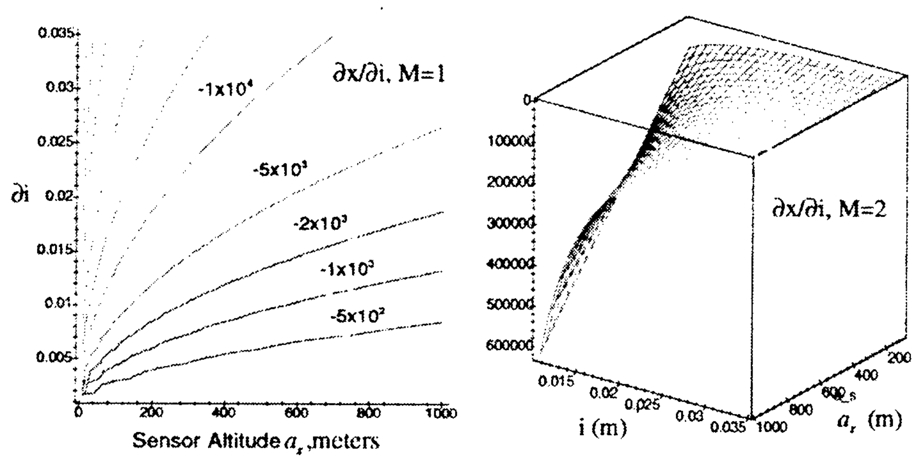


Figure 6. Proi ction of focal-plane error to the target plane at (a) M = 1 and (b) M = 2.

Setting f = 35mm and M = 1, as before, and varying a_s from 0.1km to 1km and i from 0.01f to 1.0f, we obtain the result shown in Figure 6a. Within the paraxial assumption, the preceding equation reduces to $\partial x \approx \partial i \cdot Mx/f$. Setting M = 2, we obtain the following result shown in Figure 6b:

$$\left.\frac{\partial x}{\partial i}\right|_{M=2} = \frac{f \cdot a_s \cdot \left(1 + \cot\left(\frac{\tan^{-1}(i/f)}{2}\right)\right)}{2(f^2 + i^2)} .$$

Example. Given the preceding sensor parameters, in the worst case at unitary magnification (Reference Figure 6a), a focal-plane error of $\pm 15\mu$ yields a spatial error in the target plane computed from Equation (V) as

$$\partial x = \partial i \cdot \left.\frac{\partial x}{\partial i}\right|_{a_s = 400\text{m}} = \pm 0.075\text{m} = \pm 15\mu \cdot -5 \cdot 10^3 .$$

In practice, we currently estimate that additional effects (e.g., atmospheric PSF, camera noise, reflections, etc.) could increase $\partial x / \partial i$ by a factor of two to three, thereby rendering ∂x sufficiently large to corrupt high-resolution target reconstruction. Although not a significant factor in early research in battlefield MLI (due to assumptions of low spatial resolution), such errors could severely degrade reconstruction of higher-resolution target models.

The preceding error analysis is required for parameterizing image formation and reconstruction processes. In the image formation step, a knowledge of $\partial i / \partial x$ is required due to sensor positioning and aiming errors that are projected to the focal plane. During the selection of a target reconstruction technique, the focal plane error ∂i directly influences the accuracy of target-plane projection, as shown above.

We next discuss projective errors inherent in the three-dimensional model of Figure 2, as well as practical implications of such errors for low-altitude airborne sensing.

Theorem 6. *Given Equation (IV) and the geometry of Figure 2, the projection S of a point* $\mathbf{x} \in domain(\mathbf{c})$ *to a point* $\mathbf{y} \in domain(\mathbf{d})$ *is given by*

$$\mathbf{x} = S(\mathbf{y}) = \left(\frac{a_s - a_c}{a_s}\right) \cdot h' \cdot (\sin\theta_v, \cos\theta_v) ,$$

where $h' = ([p_1(\mathbf{y})]^2 + [p_2(\mathbf{y})]^2)^{1/2}$ *and* $\theta_v = \tan^{-1}(p_2(\mathbf{y})/p_1(\mathbf{y}))$, *with* $p_k(\mathbf{y})$ *denoting projection to the k-th coordinate.*

Proof. The proof follows obviously from the givens and spherical trigonometry. ■

From Theorem 6 as well as Figures 1 and 2, the projection of a point $\mathbf{x} \in \mathbf{X}$ to a point (i,j) in the focal plane of a sensor (or camera) having magnification M, focal length f, and an anterior nodal point at a_s is given by

$$(i, j) = f \cdot \tan\left(M \cdot \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{p_1(\mathbf{x})^2 + p_2(\mathbf{x})^2}}{a_s}\right)\right]\right) \cdot (\cos\theta, \sin\theta) , \qquad \text{(VI)}$$

where $\theta = \tan^{-1}(p_2(\mathbf{x})/p_1(\mathbf{x}))$.

Error analysis. With the exception of θ, which is computed from **x**, the analysis of error in i and j proceeds similarly to the one-dimensional case, since Equation (VI) merely portrays a rotation through angle θ of the projection given in

From basic trigonometric identities, Equation (VI) reduces to

$$(\mathrm{i},\mathrm{j}) = \frac{\mathrm{f}\cdot\tan\left(\mathrm{M}\cdot\left[\frac{\pi}{2}-\tan^{-1}\left(\frac{\sqrt{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}}{a_s}\right)\right]\right)}{\sqrt{1+(p_2(\mathbf{x})^2/p_1(\mathbf{x})^2)}}\cdot\left(1,\frac{p_2(\mathbf{x})}{p_1(\mathbf{x})}\right). \qquad \text{(VII)}$$

Letting $\mathrm{v} = \tan\left(\mathrm{M}\cdot\left[\frac{\pi}{2}-\tan^{-1}\left(\frac{\sqrt{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}}{a_s}\right)\right]\right)$, the projective error components of i in Equation (VII) are given by

$$\frac{\partial \mathrm{i}}{\partial p_1(\mathbf{x})} = \frac{\mathrm{f}\cdot p_2(\mathbf{x})^2\cdot\mathrm{v}}{p_1(\mathbf{x})^3\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)^{3/2}} - \frac{\mathrm{fM}\cdot p_1(\mathbf{x})\cdot(1+\mathrm{v})}{a_s\cdot\left(1+\frac{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}{a_s^1}\right)^{3/2}\cdot\sqrt{(p_1(\mathbf{x})^2+p_2(\mathbf{x})^2)\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)}}$$

and, symmetrically for the second coordinate of **x**, we have

$$\frac{\partial \mathrm{i}}{\partial p_2(\mathbf{x})} = \frac{\mathrm{f}\cdot p_2(\mathbf{x})^2\cdot\mathrm{v}}{p_1(\mathbf{x})^2\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)^{3/2}} - \frac{\mathrm{fM}\cdot p_2(\mathbf{x})^2\cdot(1+\mathrm{v})}{a_s\cdot\left(1+\frac{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}{a_s^1}\right)^{3/2}\cdot\sqrt{(p_1(\mathbf{x})^2+p_2(\mathbf{x})^2)\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)}}.$$

We similarly obtain the error components of j as

$$\frac{\partial \mathrm{j}}{\partial p_1(\mathbf{x})} = \mathrm{f}\cdot p_2(\mathbf{x})\cdot\left(\frac{\mathrm{v}\cdot p_2(\mathbf{x})^2}{p_1(\mathbf{x})^3\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)^{3/2}} + \frac{\mathrm{v}}{p_1(\mathbf{x})\cdot p_2(\mathbf{x})\cdot\sqrt{1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}}} - \frac{\mathrm{M}\cdot(1+\mathrm{v}^2)}{a_s\cdot\left(1+\frac{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}{a_s^1}\right)^{3/2}\cdot\sqrt{(p_1(\mathbf{x})^2+p_2(\mathbf{x})^2)\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)}}\right)$$

and

$$\frac{\partial \mathrm{j}}{\partial p_2(\mathbf{x})} = \mathrm{f}\cdot p_2(\mathbf{x})\cdot\left(\frac{\mathrm{v}\cdot p_2(\mathbf{x})}{p_1(\mathbf{x})^3\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)^{3/2}} + \frac{\mathrm{v}}{p_1(\mathbf{x})\cdot p_2(\mathbf{x})\cdot\sqrt{1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}}} - \frac{\mathrm{M}\cdot(1+\mathrm{v})^2\cdot p_2(\mathbf{x})}{a_s\cdot\left(1+\frac{p_1(\mathbf{x})^2+p_2(\mathbf{x})^2}{a_s^1}\right)^{3/2}\cdot\sqrt{(p_1(\mathbf{x})^2+p_2(\mathbf{x})^2)\cdot\left(1+\frac{p_2(\mathbf{x})^2}{p_1(\mathbf{x})^2}\right)}}\right).$$

The preceding error analyses lead to two important observations, namely: (1) the forward projection errors are stable over the postulated operational range $400\text{m} \le a_s \le 800\text{m}$ and $600\text{m} \le x \le 2\text{km}$, and (2) the error function graphed in Figure 2b could be employed in constraining selection of near-optimal sensor configuration for image acquisition at low error figure in a variety of operation scenarios.

With regard to observation 1), above, the stability of a given projection error function is determined from the slope of the error surface (e.g., as shown in Figure 2b). Only in the range $a_s \le 200\text{m}$ or $x \le 400\text{m}$ is projection error significant. Note that the region where $x < a_s$ is an artifact and can be neglected since it is not physically feasible.

Concerning observation 2), above, within the operational range of altitude and range-to-target values where $x \ge a_s$, we observe that $\partial i / \partial x$ is sufficiently smooth to be encoded either as a low-order polynomial or in a lookup table of sufficient resolution. Hence, the transcendental operations involved in the computation of Equations (I) and (II) could be avoided. This could render the estimation of image acquisition error feasible on fast DSPs or small parallel processors with few floating-point capabilities. For example, the trajectory of a small airborne vehicle could thus be constrained in real time, such that imaging error $\partial i / \partial x$ as a function of range could be held to near-optimal values.

In contrast to the preceding observation, note that Figure 3 illustrates back-projection error (e.g., projection of focal-plane error to the target plane) to increase with decreasing altitude and displacement in the focal plane. In figure 3b, the surface slope is nearly linear from i = 0.015 to i = 0.02, for $500\text{m} \le a_s \le 1\text{km}$. Thus, it is reasonable to conclude that, by merging the graphs of Figures 2b and 3b, one can infer that a sensor altitude of $400\text{m} \le a_s \le 600\text{m}$ for $3a_s \le x \le 1.5a_s$ is near optimal for image acquisition and target reconstruction, under the previously-mentioned sensor configuration.

It is often useful to employ a back-projection technique for error analysis, whereby errors in the camera's focal plane or optical system can be referred to the target plane. This is especially germane to target reconstruction, where it may be desirable to determine the error in the reconstructed target that results from focal-plane quantization effects.

From Theorem 6 and Figure 2, the projection to a point $\mathbf{x} \in \mathbf{X}$ of a point (i,j) in the focal plane of a sensor (or camera) having magnification M, focal length f, and an anterior nodal point at a_s is given by

$$\mathbf{x} = a_s \cdot \tan\left(\frac{\sqrt{i^2 + j^2}}{f \cdot M}\right) \cdot \left(\sin\left(\frac{j}{i \cdot M}\right), \cos\left(\frac{j}{i \cdot M}\right)\right), \qquad \text{(VIII)}$$

<u>Error Analysis.</u> Differentiating $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$, $\mathbf{x} \in \mathbf{X}$, with respect to i and j, we obtain the following expressions for error in a two-dimensional target plane:

$$\frac{\partial p_1(\mathbf{x})}{\partial \mathrm{i}} = \frac{a_s \cdot \left(\mathrm{i}^3 \mathrm{M} \cdot \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) + \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) \cdot (\mathrm{j}\mathrm{i}^2 + \mathrm{j}^3)\right)}{\mathrm{f} \cdot \mathrm{M}^2 \mathrm{i}^2 \cdot \sqrt{\mathrm{i}^2 + \mathrm{j}^2}}$$

$$\frac{\partial p_1(\mathbf{x})}{\partial \mathrm{j}} = \frac{a_s \cdot \left(\mathrm{jMi} \cdot \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) - \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) \cdot (\mathrm{i}^2 + \mathrm{j}^3)\right)}{\mathrm{f} \cdot \mathrm{M}^2 \mathrm{i} \cdot \sqrt{\mathrm{i}^2 + \mathrm{j}^2}}$$

$$\frac{\partial p_2(\mathbf{x})}{\partial \mathrm{i}} = \frac{a_s \cdot \left(\mathrm{i}^3 \mathrm{M} \cdot \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) - \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) \cdot (\mathrm{j}\mathrm{i}^2 + \mathrm{j}^3)\right)}{\mathrm{f} \cdot \mathrm{M}^2 \mathrm{i}^2 \cdot \sqrt{\mathrm{i}^2 + \mathrm{j}^2}}$$

$$\frac{\partial p_2(\mathbf{x})}{\partial \mathrm{j}} = \frac{a_s \cdot \left(\mathrm{jMi} \cdot \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) + \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) \cdot (\mathrm{i}^2 + \mathrm{j}^3)\right)}{\mathrm{f} \cdot \mathrm{M}^2 \mathrm{i} \cdot \sqrt{\mathrm{i}^2 + \mathrm{j}^2}} .$$

4.3. Optical and Media Errors

In the simulation of realistic optical effects, errors arise due to uncertainty in camera parameters such as focal length or magnification. Additional error sources in MLI simulation and error analysis models comprise inaccurate estimation of media optical parameters, such as the scattering or absorption coefficient. We discuss such errors in the following development.

In Equation (VII), if the lens focal length or magnification is erroneously specified or measured, then taking the derivative of $\mathbf{x} \in \mathbf{X}$ with respect to f or M yields the following optical system errors, expressed vectorially:

$$\frac{\partial \mathbf{x}}{\partial \mathrm{f}} = a_s \cdot \frac{\sqrt{\mathrm{i}^2 + \mathrm{j}^2}}{\mathrm{f}^2 \mathrm{M}} \cdot \left(\cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right), \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right)\right)$$

and

$$\frac{\partial \mathbf{x}}{\partial \mathrm{M}} = a_s \cdot \frac{\sqrt{\mathrm{i}^2 + \mathrm{j}^2}}{\mathrm{f} \cdot \mathrm{M}^2 \mathrm{i}} \cdot \left(\mathrm{i} \cdot \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) - \mathrm{j} \cdot \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right), \ \mathrm{i} \cdot \sin\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right) + \mathrm{j} \cdot \cos\left(\frac{\mathrm{j}}{\mathrm{i} \cdot \mathrm{M}}\right)\right) .$$

Lens errors and distortions are not the only optical effects that can degrade the accuracy of target reconstruction. For example, media effects such as scattering and absorption can corrupt the received image, thereby inducing spatial and greyscale perturbations in the focal plane when media optical parameters are erroneously prespecified or inaccurately measured. In practice, due to errors resulting from current in situ sensor technology, media optical parameters can only be estimated. Hence, it is useful to consider the error involved in modelling optical effects.

Let a volume scattering function $\beta : (-\pi, \pi] \to [0, 1]$ generate the weights of a template $\mathbf{s} : \mathbf{R}^+ \times \mathbf{X} \to \mathbf{R}^{\mathbf{X}}$ that instantiates a linear, single-scattering, near-field approximation of β. Denoting the terminus of a scattering process' optical axis as $\mathbf{y} \in \mathbf{X}$ and the terminus of a ray scattered off-axis as $\mathbf{x} \in \mathbf{X}$, a scattering template **s** is defined in terms of its weights as:

$$\mathrm{s}_{\mathbf{y}}(d)(\mathbf{x}) = \beta(\vartheta), \text{ where } \vartheta = \tan^{-1}\left(\frac{\| \mathbf{y}-\mathbf{x} \|}{d}\right) \text{ with } d \in \mathbf{R}^{+} \text{ and } \mathbf{x}, \mathbf{y} \in \mathbf{X} . \qquad \text{(IX)}$$

Error Analysis. It is possible that the scattering length d can be incorrectly estimated from the corresponding scattering coefficient b, since the mean value of d for a volume scattering function β is derived as $d = 1/\mathrm{b}$. Error analysis of the template definition in Assumption 2.4.3 proceeds as follows. Letting $\mathrm{v} = \| \mathbf{y}-\mathbf{x} \|$, the error $\partial \mathrm{s}_{\mathbf{y}}(\mathbf{x})$ in a given template weight due to a perturbation $\partial \mathrm{b}$ in the scattering coefficient is given by

$$\frac{\partial \mathrm{s}_{\mathbf{y}}(\mathbf{x})}{\partial \mathrm{b}} = \frac{\mathrm{D}(\beta) \cdot \mathrm{v}^{1/2} \cdot \tan^{-1}(\mathrm{v}^{1/2} \cdot \mathrm{b})}{1+\mathrm{b}^2\mathrm{v}} ,$$

where $\mathrm{D}(\beta)$ denotes the derivative of $\beta(\vartheta)$, at ϑ corresponding to **x** and **y**, per Assumption 2.4.3. Within the near field of β, the approximation $x = \tan(x)$ holds, which yields the following simplification:

$$\frac{\partial \mathrm{s}_{\mathbf{y}}(\mathbf{x})}{\partial \mathrm{b}} \approx \frac{\mathrm{D}(\beta) \cdot \mathrm{vb}}{1+\mathrm{b}^2\mathrm{v}} = \frac{\mathrm{D}(\beta) \cdot \mathrm{vb}}{\mathrm{v}(\mathrm{v}^{-1}+\mathrm{b}^2)} = \frac{\mathrm{D}(\beta) \cdot \mathrm{b}}{\mathrm{v}^{-1}+\mathrm{b}^2} .$$

This indicates that the error due to scattering varies inversely with b and directly with v, the square of the distance between **x** and **y**. In practice, this means that the intensity error in the scattering template is smaller for turbid media (larger b), which directly implies shorter scattering distances.

The spatial effects of scattering are similarly described. For example, the scattering angle ϑ can be expressed as $\vartheta = \tan^{-1}(\sqrt{\mathrm{v}/\mathrm{d}}) = \tan^{-1}(\sqrt{\mathrm{bv}})$, which yields the result $\vartheta = \sqrt{\mathrm{bv}}$. Hence, the displacement perpendicular to the optical axis is given by $\mathrm{v} = \vartheta^2/\mathrm{b}$, which exhibits the following dependencies upon b and ϑ:

$$\frac{\partial \mathrm{v}}{\partial \mathrm{b}} = \left(\frac{\vartheta}{\mathrm{b}}\right)^2 \quad \text{and} \quad \frac{\partial \mathrm{v}}{\partial \vartheta} = \frac{2\vartheta}{\mathrm{b}} .$$

Note that $\partial \mathrm{v}$ increases supralinearly as b decreases and ϑ increases. Hence, narrow-angle scattering in turbid water would yield more accurate simulation of lateral displacement due to scattering effects. This is consonant with our preceding assumption of the near-field approximation.

Absorption is generally characterized by the coefficient a, which yields attenuation α over a propagation distance d that is given by $\alpha = e^{-\mathrm{a}d}$.

Error Analysis. Given the preceding observation, taking the derivatives of α with respect to a and d, we obtain

$$\frac{\partial \alpha}{\partial \mathrm{a}} = -d \cdot e^{-\mathrm{a}d} \quad \text{and} \quad \frac{\partial \alpha}{\partial d} = -\mathrm{a} \cdot e^{-\mathrm{a}d} .$$

In practice, this means that the error involved in estimating the absorption effect decreases with increasing a and d, since the exponential function predominates.

In practice, modelling of scattering and absorption is performed by linear convolution and multiplication operations. For example, let $\mathbf{b} \in \mathbf{R}^{\mathbf{X}}$ denote an undistorted image whose pixels have been formed by a projective process that is applied concurrently with scattering and absorption processes. Given template s defined in Equation (IX) and the absorption constant a, together with an array of propagation distances $d \in (\mathbf{R}^{+})^{\mathbf{X}}$, single scattering and absorption are simulated as follows:

$$\mathbf{c} = (\mathbf{b} \cdot e^{-\alpha d}) \oplus \mathbf{s}(d) \ .$$

This approximation has been verified as physically accurate for simulating the effects of near-field forward scattering in moderately turbid seawater and atmospheres [Sch96c], and is hence useful for the purposes of this preliminary study.

4.4. Camera Errors

Physical imaging sensors generally have numerous error sources that directly and adversely impact the reconstruction of digitally imaged targets. Included in these error sources are the following primary effects:

1) *Intensifier noise* produced by avalanche phenomena inside the microchannel tubules of intensifier tubes appears to provide the primary contribution to image intensifier noise. Often described in terms of Gaussian statistics for convenience, intensifier noise has also been characterized in terms of Lorentzian distributions [Sar92];
2) *Nonuniform pixel gain*, which tends to decrease toward the detector periphery, resulting in vignetting of the majority of non-zero or non-saturated image frames output by a digital camera;
3) *Pixel errors* such as missing pixels or (infrequently) missing rows and columns of pixels; and
4) *Preamplifier noise*, which tends to exhibit Gaussian or Poisson statistics as a function of spatial coordinate.

Camera errors can thus be concisely described by a noise map $\mathbf{b}_n$, gain map $\mathbf{b}_g$ and pixel defect map $\mathbf{b}_p$. For example, if a noiseless (synthetic) camera image $\mathbf{a}$ is a real-valued image on an $M_c \times N_c$-pixel domain $\mathbf{X}$, then $\mathbf{b}_n, \mathbf{b}_g, \mathbf{b}_p \in \mathbf{R}^{\mathbf{X}}$ and, assuming additive noise, the simulated (noise-corrupted) image $\mathbf{d}$ is given by:

$$\mathbf{d} = \mathbf{b}_p \cdot \mathbf{b}_g \cdot (\mathbf{a} + \mathbf{b}_n) \ ,$$

where $\mathbf{b}_p$ and $\mathbf{b}_g$ are prespecified and can correspond to a given type of detector. In contrast, $\mathbf{b}_n$ is specified in terms of a Gaussian distribution about a random variable $r(\mathbf{x}) \in \mathbf{R}, \mathbf{x} \in \mathbf{X}$, as

$$\Pr(r(\mathbf{x})) = \frac{e^{-(r(\mathbf{x}) - \mu)^2/(2\sigma)^2}}{\sqrt{2\pi} \cdot \sigma} \ ,$$

where μ and σ signify the mean and standard deviation, respectively, or by a Lorentzian distribution

$$\Pr(r(\mathbf{x})) = \frac{\alpha}{\pi(r^2(\mathbf{x}) + \alpha^2)} \ .$$

4.5. Glint Effects

When a target or its background exhibits specular reflectance (e.g., glint from a vehicle windshield or the corrugated surface of a wind-rippled pond), such reflectance effects can enhance or degrade target imagery. For example, glint from metallic parts on a vehicle disguised by partially opaque camouflage (e.g., netting) can provide structured information regarding key target features. Similarly, glint from background proximal to a camouflaged target could provide information about possible target presence and shape (e.g., in non-glint areas).

Alternatively, glint collocated with or surrounding small targets can cause saturation proximal to target pixels in a received image. This instance of glint has the usual effect of reducing signal-to-noise ratio in the neighborhood of the glint pixels, due to inter-channel crosstalk. Such phenomena tend to be amplified by intensifier tubes, as discussed in Reference [Sch96d].

In the computer model developed for this study, glint is simulated as follows:

Step 1. Assume that there exists a light source at point $\mathbf{p}_L = (x_L, y_L, z_L)$ and an imaging sensor at $\mathbf{p}_i = (x_i, y_i, z_i)$.

Step 2. For each target point $\mathbf{x} = (x_t, y_t, z_t) \in \mathbf{X}$, compute the angles δ and ξ as

$$\delta = \tan^{-1}\left(\frac{\sqrt{(x_L - x_t)^2 + (y_L - y_t)^2}}{z_L - z_t}\right) \quad \text{and} \quad \phi = \tan^{-1}\left(\frac{\sqrt{(x_i - x_t)^2 + (y_i - y_t)^2}}{z_i - z_t}\right) .$$

Step 3. Compute glint propagation angle as $\xi = |\delta - \phi|$.

Step 4. Assuming the existence of a radially symmetric angular reflectance (glint) response function $\gamma : (-\pi, \pi] \rightarrow [0, 1]$, determine $g_g = \gamma(\xi)$, which is the gain due to glint at target point $\mathbf{x} \in \mathbf{X}$.

Step 5. Given a target reflectance map $\mathbf{r} \in [0, 1]^{\mathbf{X}}$, and a light source of intensity I_s, compute the glint image as

$$\mathbf{g} \equiv \{(\mathbf{x}, \mathbf{g}(\mathbf{x})) : \mathbf{g}(\mathbf{x}) = I_s \cdot g_g(\mathbf{x}) \cdot \mathbf{r}(\mathbf{x}), \mathbf{x} \in \mathbf{X}\} . \tag{X}$$

Observe that several simplifying assumptions have been implemented in the preceding algorithm, as follows:

1) A spatially uniform reflectance function γ is assumed, which does not occur in practice, due to spatial variance in composition of scene materials. For example, the specular reflectance of a vehicle's windshield is more sharply peaked about $\xi = 0$ that the generally Lambertian reflectance of treetops or a grassy field sampled at spatial scales ranging from 1m to 10m.

2) The reflectance angle ξ is not computed with respect to a non-horizontal target or background surface normal at a given target point $\mathbf{x} \in \mathbf{X}$. Although the ensemble average (i.e., a horizontal target or background surface) is assumed, the majority of surface normals are non-vertical, especially in foliated scene neighborhoods.

3) A point light source is assumed, and is implemented by not accounting for illuminant divergence, as well as camera admittance angle. This simplifying assumption reflects the emphasis of this study on first- and second-order effects, namely, specular reflectance in the case of glint effects.

We next discuss the rationale for each of the foregoing assumptions.

Spatial nonuniformity of the surface reflectance angular response, while realistic, can be difficult to simulate in an optical effects model not specifically associated with, or compatible with, a 3-D computer graphics or rendering model or software package. In particular, a 3-D model customarily specifies target and background objects in terms of an ensemble of spatially disjoint facets, each of which can have a surface normal that is distinct from the surface normals of spatially adjacent facets. Although we have shown that this simplified modelling approach facilitates accurate approximation of continuous surfaces by finite discrete models, the computational overhead required to render such views can be prohibitively large. In this study, we emphasize the fast modelling of different MLI scenarios, for purposes of comparing and contrasting target visibility, as well as providing large databases for follow-on error analysis. The usual rendering cost was not acceptable under such constraints, and we chose instead the simpler (but statistically accurate) method of a uniform reflectance response function referenced to the ensemble average of surface slopes. In future versions of this model, we plan to introduce the more realistic treatment afforded by a rendering model, as time and support permits.

The assumption of a point light source is acceptable for active-source illuminant models, and as a first approximation to insolation (*in*coming *sol*ar radi*ation*) in the presence of a clear atmosphere. Since the vast majority of specular reflectance effects occur in the presence of point or extended-point sources, the point-source assumption is reasonable for this first-order model. We plan to extend this to extended-point and diffuse sources, as time permits.

4.6. Effect of Cover Aperture Size and Spacing

The spatial extent, interstitial spacing, and transmittance or cover objects can vary widely in practice. In the initial part of this study, we examined the effects of cover object size and spacing (e.g., tree diameter and distance between centers of tree trunks), attempting to determine if there was a correlation between cover object size and separation and target visibility. What was discovered in later parts of this study was that the cover *aperture* size and spacing was the chief determinant of target visibility in a given sensing scenario. In particular, determining the overlap effect of successive views or looks as a function of aperture parameters proved to be a more effective analytical perspective.

For example, consider the circular aperture shown in Figure 7a. Given a linear track with views closely (widely) separated in space, the composite aperture effects shown in Figure 7b (7c) would result, all other parameters being held constant. Qualitative observation notes that wide spacing of small apertures tends to offer little or no advantage in terms of obtaining multiple views per given target pixel, but nevertheless provides greater total coverage in the target plane across all views than would be obtained by any single view. Given the geometry of Figure 1, it is possible to estimate the interframe distance along a sensor track required to achieve minimal overlap, as follows.

Let circular apertures of diameter g_c meters be located at an altitude a_c meters above the target plane, over which a sensor flies at altitude a_s meters along a linear or quasi-linear track that crosses the centers of the apertures, as shown in Figure 8a. Here, the assumption of bisection is made to simplify the following analysis, and states the ensemble average. Assuming nadir viewing to minimize the keystoning effect, observe (from Figures 1 and 8b) that the cover aperture width projected to a flat target plane is given by the following expression for the mean:

$$\bar{g}_t = \frac{a_s}{a_s - a_c} \cdot \bar{g}_c \, . \tag{XI}$$

It is readily determined by inspection of Figure 8b that the mean interframe distance along a sensor track that the mean interframe distance along the sensor track required to achieve frame overlap (and, hence, multiple views of at least n-1 target pixels given n sensor positions) is given by

$$d_{if} \leq \bar{g}_t \, .$$

The interframe overlap effect is closely related to sensor speed s and frame rate F, as shown in the following equation:

$$F = \left\lceil \frac{s}{d_{if}} \right\rceil .$$

The following example is illustrative.

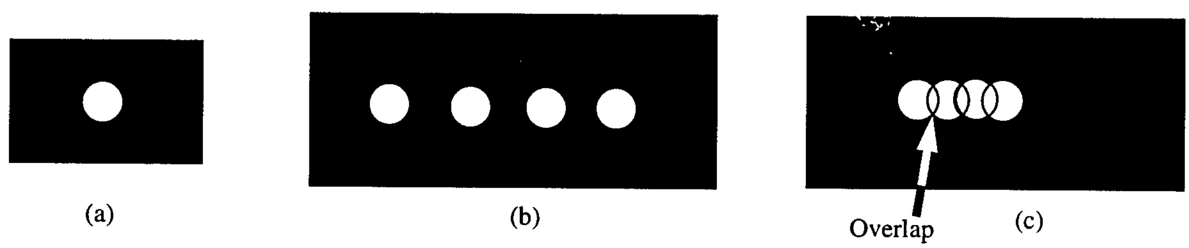


Figure 7. Effect of aperture separation: (a) simulated cover, (b) widely spaced apertures projected to the target plane, (c) closely spaced apertures projected to the target plane.

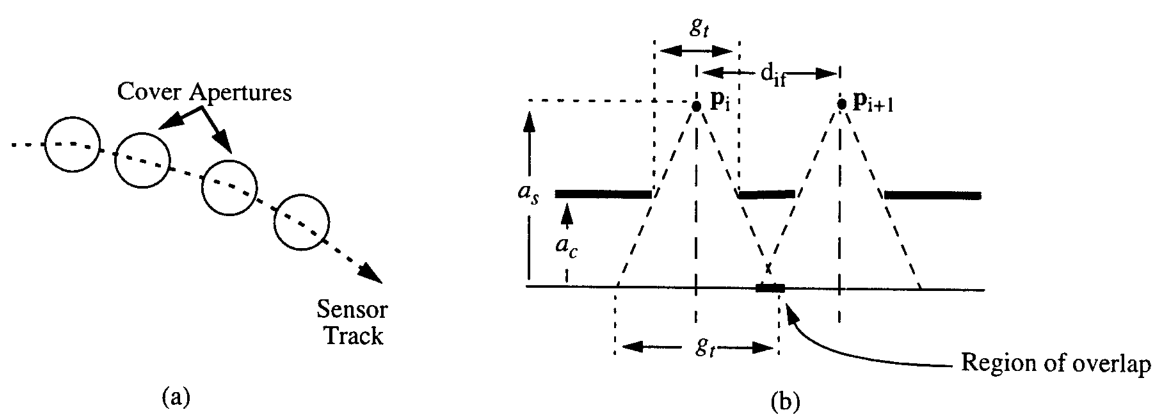


Figure 8. Geometry of multi-view overlap in successive frames: (a) sensor track crosses centers of circular apertures, and (b) sensing geometry for determining interframe distance d_{if} to achieve interframe overlap.

Example. Let a_s = 1km, a_c = 20m, and $\bar{g}_c$ = 10m. Then, d_{if} = 1000(10)/980 = 10.2 meters. Assuming a sensor velocity across the midpoints of adjacent apertures having speed component of s = 200 mph = 94.1 meters/second, the frame rate required to achieve this interframe distance would be at least

$$F = \lceil s/d_{if} \rceil = \lceil 94.1\text{m/s}/10.2\text{m} \rceil = 10 \text{ frames/sec}.$$

Error Analysis. Taking the derivative of Equation (XI) with respect to a_c and a_s yields:

$$\frac{\partial \bar{g}_t}{\partial a_c} = \frac{a_s}{(a_s - a_c)^2} \cdot \bar{g}_c \quad \text{and} \quad \frac{\partial \bar{g}_t}{\partial a_s} = -\frac{a_c}{(a_s - a_c)^2} \cdot \bar{g}_c ,$$

which means that as a_s approaches a_c, the preceding error measures will become unstable. In practice, this indicates that interframe overlap can increase drastically as the sensor platform approaches the cover from above (e.g., flies just above the treetops). However, there also exists the practical problem that objects in the target plane are occluded from oblique viewing by the sensor by structures such as tree trunks, brush, etc. This effect is discussed in the following paragraphs, and constitutes a significant complication in the simulation of low-altitude MLI.

An important consideration in MLI with linear tracks is the effect of target occlusion by three-dimensional cover objects such as the vertical or nearly-vertical treetops of coniferous forests (as illustrated schematically in Figure 9a) as opposed to the spherical or pear-shaped cross-sectional profiles of deciduous trees (Figure 9b). We have found that cover objects with flatter vertical cross-sectional profiles (e.g., reduced height-to-diameter ratio r_{hd}) tend to hide ground targets less well than generally conical profiles of coniferous trees (larger r_{hd}), other parameters being equal.

In Equation (XI), the variable $\bar{g}_c$ is assumed to be measured along-track. It would appear to a naive observer that interframe overlap could be maximized when the sensor track follows the projection of the longest cover segments. However, this is not necessarily true, due to 3-D effects incurred by the presence of projecting cover objects that are constituent to the cross-sectional geometry of various cover objects. for example, as shown in Figure 9c, it may be advantageous for sufficiently agile sensors to descend into the region of target visibility delimited by the projection of cover boundaries upward from target objects to the sensor track. In contrast, our development in this report has assumed projection of cover segments downward to the target plane from the reference location of the imaging sensor's nodal point, in order to determine target visibility. However, when determining optimal sensor track, reverse projection is useful, because it provides information about regions of target visibility into which an airborne sensor could fly, to obtain optimal target imagery (i.e., with significantly reduced probability of occlusion). This concept is further discussed in Sections 5.1, 5.2, and 6.3, after we present simulation methodology and examples.

4.7. Target Visibility and Error Simulation Model

As shown in Figure 10, the concept of MLI simulation in the context of this study is to combine multiple views of an object to determine (a) target visibility and (b) error of estimation associated with the visibility estimate. The latter measure is especially important when determining the potential accuracy of target reconstruction methods, such as presented in Section 5.

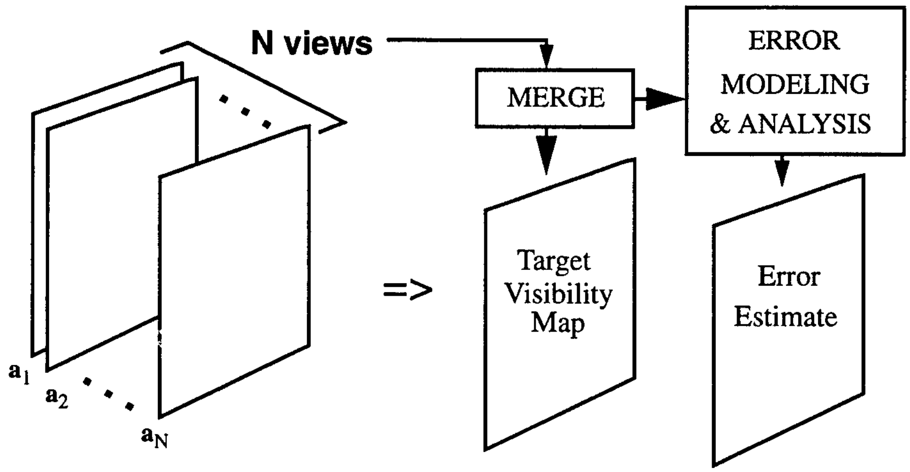


Figure 10. MLI simulation concept overview, where the *merge* step denotes a target visibility estimation algorithm.

We have constructed a simulation model in computer software that estimates a target visibility map as well as an error budget for MLI given user-specified sensor track, cover configuration and altitude, atmospheric optical parameters, camera noise and pixel defect parameters, and glint effects. This model has the structure shown in Figure 11. observe the parallel functionality for image simulation and error estimation, which facilitates the verification of target visibility information at a given point $\mathbf{x} \in \mathbf{X}$ of the target plane by consulting the error estimation map at $\mathbf{x}$. Additionally, the modular structure of this model facilitates maintenance and expansion to meet future research needs.

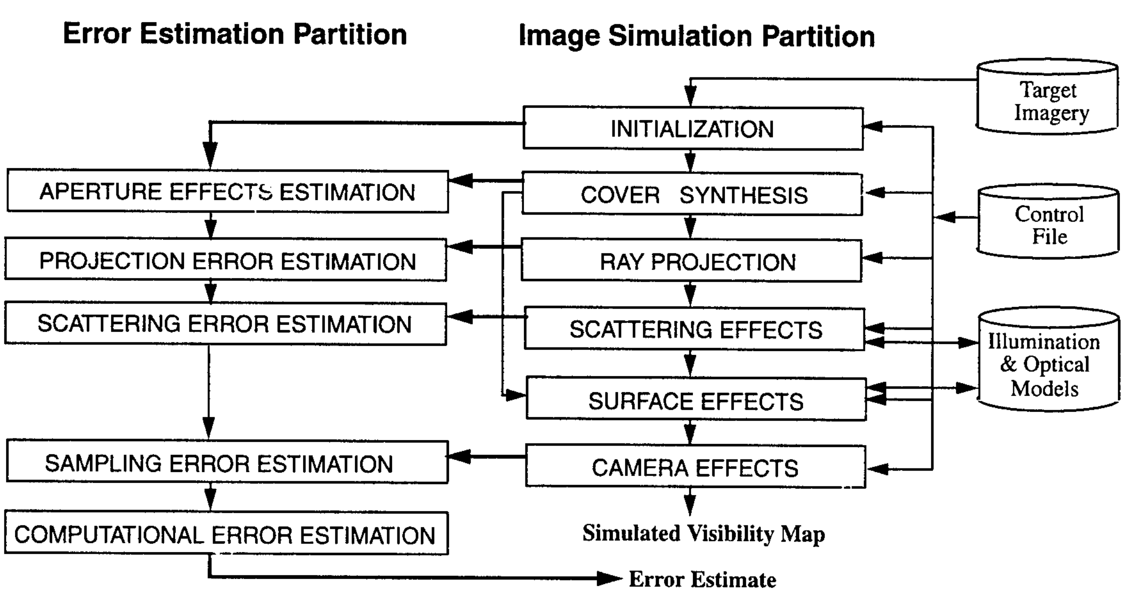


Figure 11. MLI simulation model architecture, showing concurrent error estimation and image simulation.

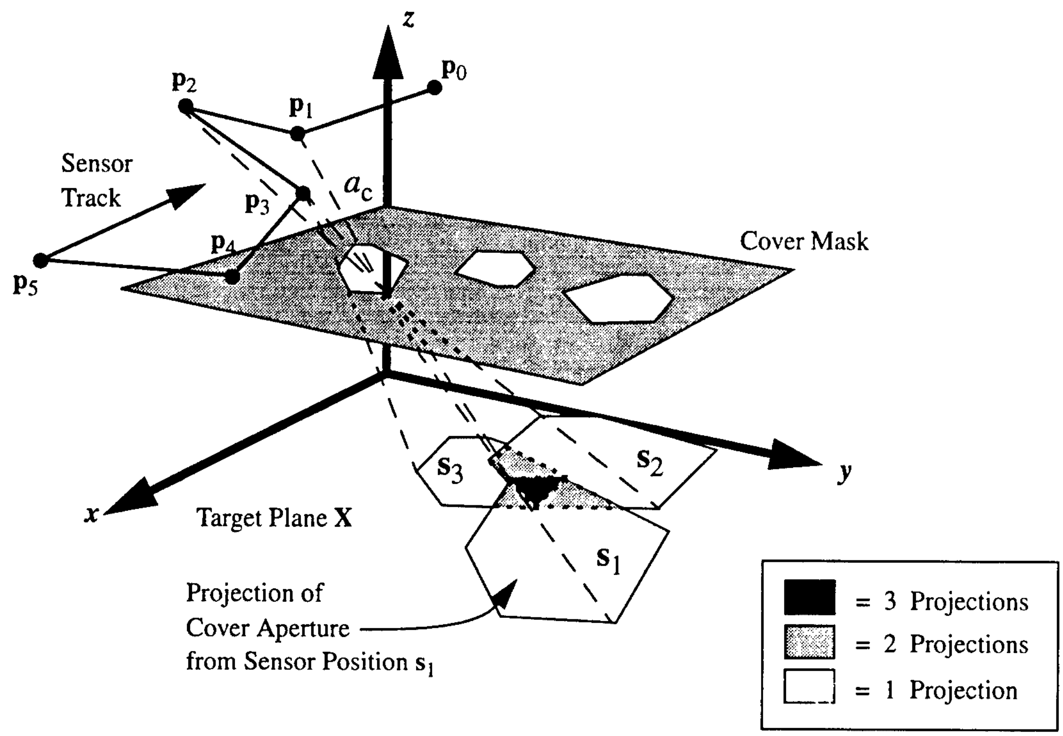


Figure 12. Schematic diagram of projection process that forms target visibility map.

Target visibility is simulated by projecting each sensor pixel boundary at each sensor position $\mathbf{p}_i$, i = 0..n-1, to the target plane. The accumulation of cover transmittance adjusted for atmospheric effects comprises the basic target visibility map. This process is illustrated schematically in Figure 12. Error maps are produced that account for camera and glint effects within the accuracy of approximations discussed in Sections 4.4 through 4.6. Such error maps can be combined with the basic target visibility map, for analytical purposes, as discussed in Section 5.

Algorithm. The basic projection algorithm proceeds as follows. Given an MxN-pixel cover image $\mathbf{c} \in [0, 1]^{\mathbf{Y}}$, where M and N are prespecified; a sensor track comprised of a sequence of three-dimensional points $P = \{\mathbf{p}_0, \mathbf{p}_2, \ldots, \mathbf{p}_k, \ldots, \mathbf{p}_{n-1}\}$, where $\mathbf{p}_k = (x_k, y_k, z_k)$; resolution intervals at the cover or target planes $\Delta x_c, \Delta y_c$ or $\Delta x_t, \Delta y_t$, with cover and altitudes denoted by $z_c, z_t \in \mathbf{R}^+$, perform the following steps:

Step 1. Initialize **b**, **c** := **0**, the zero-valued constant image.

Step 2. **for** k = 0 **to** n-1, **do**:

(2a) **if** $\Delta x_t, \Delta y_t$ are unspecified, **then** $\Delta x_t = \Delta x_c \cdot z_s/(z_s - z_c)$ and $\Delta y_t = \Delta y_c \cdot z_s/(z_s - z_c)$

(2b) **for** i = 1 **to** M **do**:

(2c) $$x_c = \Delta x_c \cdot i \ ; \quad x_t = \left[(z_s - z_t) \cdot \frac{x_c - x_s}{z_s - z_c}\right] + x_s \ ; \quad i_t = nint(x_t/\Delta x_t)$$

(2d) **for** j = 1 **to** N **do**:

(2e) $y_c = \Delta y_c \cdot j$; $y_t = \left[(z_s - z_t) \cdot \frac{y_c - y_s}{z_s - z_c}\right] + y_s$; $j_t = nint(y_t / \Delta y_t)$

(2f) $\mathbf{a}(i, j) = (x_t, y_t, z_t)$; $d_s(i, j) = \| \mathbf{p}_k - \mathbf{a}(i, j) \|$

(2g) **if** i > 1 and j > 1, **then**

(2g.1) Construct a polygon (rectangle, triangle, or trapezoid) from **a**(i ,j) and the adjacent pixel vertices **a**(i-1,j), **a**(i,j-1), and **a**(i-1,j-1). Denote the points in **X** contained in this polygon by *N*(i,j).

(2g.2) Let the accumulated transmittance image **d** be augmented as $\mathbf{d}|_{N(i,j)} = \mathbf{d}|_{N(i,j)} + \mathbf{c}(i, j)$.

(2g.3) Let the accumulator image for the number of views per target pixel be augmented as $\mathbf{b}|_{N(i,j)} = \mathbf{b}|_{N(i,j)} + 1$.

(2h) **endif** ; **endfor** ; **endfor** ; **endfor**

Step 3. Calculate the mean transmittance per target pixel as **f = d / b**

Step 4. Given the scattering template **s** defined in Equation (IX) as well as scattering and absorption coefficients b and α, perturb **f** to portray the effects of scattering and absorption, as follows:

(4a) Compute the order of scattering $k_s = d_s(i, j) \cdot b$.

(4b) Let the simulated result be given by $\mathbf{h}_k = (\mathbf{f} \cdot e^{-\alpha d_s}) \overbrace{\oplus \mathbf{s}(d_s) \oplus \mathbf{s}(d_s) \oplus \ldots \oplus \mathbf{s}(d_s)}^{k_s \text{ times}}$

Step 5. Combine the individual maps $\mathbf{h}_k$, for k = 0..n-1, via pointwise summation over **X**, that is

$$\widehat{\mathbf{h}} \equiv \left\{ (\mathbf{x}, \widehat{\mathbf{h}}(\mathbf{x})) : \widehat{\mathbf{h}}(\mathbf{x}) = \sum_{k=0}^{n-1} \mathbf{h}_k(\mathbf{x}), \mathbf{x} \in \mathbf{X} \right\} .$$

Step 6. Calculate the camera noise as shown in Equation (X), to produce a camera noise map $\mathbf{n}_k \in \mathbf{R}^{\mathbf{X}}$ for each of k = 0..n-1 views. Combine all $\mathbf{n}_k$ by summation, as in Step 5, to yield a composite noise map $\widehat{\mathbf{n}}$ that portrays additive accumulation of noise between frames.

Step 7. Calculate the glint effects as shown in Section 4.5, to produce a glint effects map $\mathbf{m}_k \in \mathbf{R}^{\mathbf{X}}$ for each view. Combine the noise maps by summation, as in Step 5, to yield a composite glint map $\widehat{\mathbf{m}}$, which portrays additive accumulation of glint.

The function of each algorithm step is explained as follows.

All data stated as givens are read from a control file, or from ancillary files referenced by filename in the control file. This allows various cover configurations, noise and glint parameter sets, and sensing geometries to be specified in a flexible way. In Step 1, the counter image **b** and the transmittance accumulator image **d** are initialized to zero, which is appropriate for accumulators. Step 2 begins with the k-loop, which directs the simulation by stepping through the multiple views possible from the sensor positions in P. Step 2a computes the target-plane instantaneous field-of-view (IFOV) in the x- and y-axis directions, if such components are not specified in the control file. Step 2b iterates through the rows of pixels in the cover image, and Step 2c contains computations of the cover pixel coordinate x_c, target pixel coordinate x_t, and target pixel index i_t. Here, the *nint*(x) function computes the integer nearest the real argument x. Steps 2d and 2e are symmetric to Steps 2b and 2c, but apply to column coordinates. In Step 2f, the target-plane coordinates are stored in **a**, and this information is transformed into a propagation distance stored in d_s, which is computed between the sensor position $\mathbf{p}_k$ and the target-plane coordinate (x_t, y_t, z_t).

If processing is not occurring within the first row or column of **c**, then it is possible to construct the projection of the current cover pixel to the target plane. This is done by computing the target plane loci within the polygonal neighborhood N whose vertices are given by **a**(i,j), **a**(i-1,j), **a**(i,j-1), and **a**(i-1,j-1). In cases where three vertices are colinear, N has triangular shape, otherwise N is rectangular or trapezoidal. The determination of N usually represents the most computationally costly step of the projection algorithm, since line intersections must be computed, and a version of Bresenham's algorithm for computing the points on a straight line of arbitrary slope must be employed [Kuz95]. Once the points within the aforementioned polygon are collected in N, the updates to **b** and **d** can be efficiently computed. Step 2h is merely the termination of nested **if** and **for** loops.

The physical justification for accumulation of transmittance in **d** (Step 2g.2) may not be immediately apparent, but can readily be seen if one considers the effect of a light source of constant intensity positioned at each of the points of P, which source shines through the cover apertures. Clearly, the accumulated irradiance at any point in the target plane would be directly proportional to the accumulated cover transmittance projected to that point. Similarly, the averaging of accumulated transmittance in Step 3 may not be immediately obvious, but can be readily understood by noting that **b** contains the number of ray bundle projections through the pixels of the cover plane to a given point in the target plane. Since *range*(**b**) = [0,n], pointwise division of **d** by **b** yields the average cover transmittance at each point of **X** for the n simulated views. A task for future research would emphasize determination of whether or not the mean is the most effective measure in this situation -- for example, should the median or some statistical characterization (e.g., m-tile boundaries) be respectively employed to characterize MLI in the presence of small or large n.

The simulation of multiple scattering by repeated application of the linear model described in Section 4.3 has been a feature of our aqueous scattering models for the past eight years, and has been verified in practice as being sufficiently accurate within two to three attenuation lengths that estimation error is within the greyscale quantization interval (i.e., 0.39 percent of peak intensity) of an eight-bit digital image [Sch96c]. Since three orders of multiple scattering rarely occur in airborne imaging through clear or somewhat hazy atmospheres (where the scattering coefficient b is small in relationship to its value in aqueous media), the accuracy of this approximation for near-field forward scattering should not be a primary concern in the analysis of simulation error.

The additive combination of average transmittance maps is justified in this preliminary model by observing that the component maps $\mathbf{h}_k$ are computed via the arithmetic mean, and hence have addition as a component generative operation. The additive combination of noise or glint component maps to yield a composite map is consonant with the observation that noise tends to accumulate additively in imaging systems, particularly in intensified cameras [Sar92,Sch96e], and the preceding observation concerning additive accumulation of irradiance (of which glint is an instance).

We next present several simple examples of the analysis of simulation model output.

5. Visibility and Error Simulation and Analysis

In this section, we summarize results obtained from numerous model simulation runs and analysis of MLI-based target reconstruction algorithms. Section 5.1 discusses an example loitering scenario, while imaging along a linear track is analyzed in Section 5.2. Such discussion emphasizes different modes in which the simulation model can be employed to focus on the effect of specific physical phenomena and associated constraints. In Sections 5.3 , 5.4, and 5.5, we respectively present error analyses of target reconstruction based on cepstral coregistration, relative geometry, and tomography at unequally-spaced sensor positions.

5.1. Example Simulation: Loitering Scenario

Let a sensor comprised of an imaging device having unitary magnification, infinite depth-of-field (idealized pin-hole camera), and Gaussian noise of mean 0.1 and standard deviation 0.02 over a greyscale interval [0,1] be positioned according to the curvilinear track listed in Table 1, which is a segment of a loitering track. Simulated sensor altitude is 300m, cover altitude is 20m, and atmospheric absorption and scattering coefficients were set at a = 0.003 m^{-1} and b = 0.002 m^{-1}.The cover image and the composite visibility image are shown in Figure 13. Note how the overlay of each projected cover aperture accumulates in the composite image, where bright (dark) pixels correspond to target regions where many (few) views are available. The data of Table 2 are graphed in Figure 13c.

Table 1. Example loitering scenario: Sensor track points.

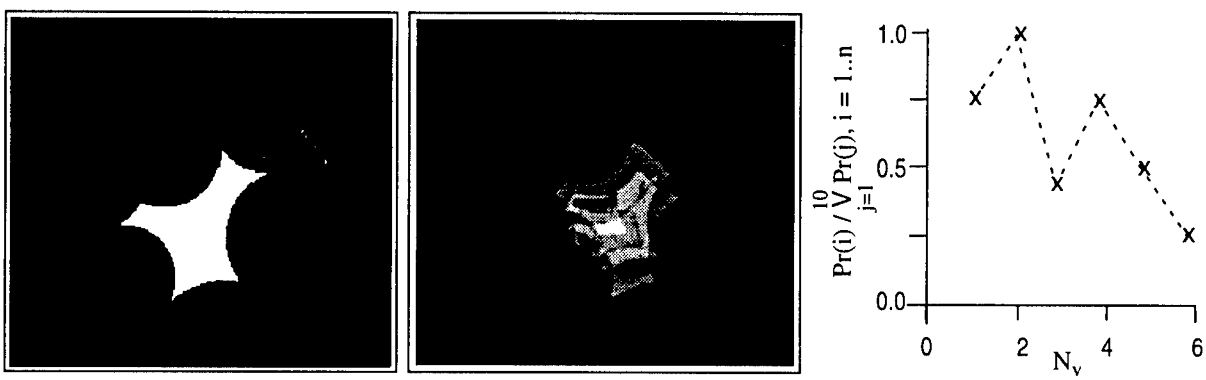
Track Point	x_s, meters	y_s, meters	z_s, meters
$\mathbf{p}_0$	228	0	300
$\mathbf{p}_1$	150	50	240
$\mathbf{p}_2$	128	100	200
$\mathbf{p}_3$	114	150	270
$\mathbf{p}_4$	177	200	290
$\mathbf{p}_5$	310	250	300

The probability of a given number of views for this example scenario is expressed in Table 2. Note that the large probability of no views (Pr(0) = 0.956) is a function of the size and resolution of **X**. For example, if the brightest region of Figure 13b, corresponding to a total of six views per pixel, contains 1,000 pixels and **X** is 1024x1024 pixels in size, then Pr(6) = 0.001. However, if **X** is increased to 2048x2048 pixels, then Pr(6) = 0.00025, all other parameters being held constant. Clearly, the inclusion of Pr(0) biases the remaining probabilities and is thus eliminated in Figure 13c, by normalizing each probability Pr(i), i = 1..n, to the maximum of Pr(i). Alternatively, one could normalize the probabilities to the sum

$$S = \sum_{j=1}^{n} \Pr(j) = 1 - \Pr(0) .$$

Table 2. Example loitering scenario: Probability of target visibility given a number of views n = 6.

Number of Projections per Pixel, N_v	Probability
0	0.956
1	0.009
2	0.012
3	0.005
4	0.009
5	0.006
6	0.003



(a) (b) (c)

Figure 13. Loitering track scenario: (a) cover mask (256x256 pixels), and (b) projections to target plane corresponding to the statistical data of Table 2, and (c) graph of probability of projection $N_v = i$ where i = 1..6, normalized for maximum non-zero projections, versus number of projections N_v

We recommend this normalization technique for future analysis, since it eliminates bias within the error of ±1 pixel incurred by sampling over the |X|-pixel target domain.

The variation in the extent of a given view with altitude is readily ascertained from inspection of Figure 1. Observe that, at infinite altitude and nadir viewing, the cover apertures would be projected directly downward to the target plane. Although this case does not occur in practice, it serves to illustrate one extreme of sensor altitude, at which

$Pr(N_v = 1)$ tends toward unity. Tending toward the other extreme, as sensor altitude approaches cover altitude, the projection of each cover aperture to the target plane becomes larger, and such projections tend to overlap increasingly. This effect increases the number of views for which $N_v > 1$, and hence, increases $Pr(N_v)$ for $N_v = 2..n$.

The increase in number of potential target views per pixel with decreasing altitude is shown in Figure 14b, for a cover mask similar to that in Figure 13, but with multiple apertures and partially overlapping circular cover segments. Cover altitude is fixed at 20m for all sensor altitudes, and the target plane has four times the resolution of the cover plane, to accommodate the expanding projections of cover apertures as sensor altitude is decreased (e.g., if the cover mask is 256x256 pixels, then the target plane has 1024x1024 pixels). Because the cover apertures overlap insufficiently, Pr(n) is vanishingly small.

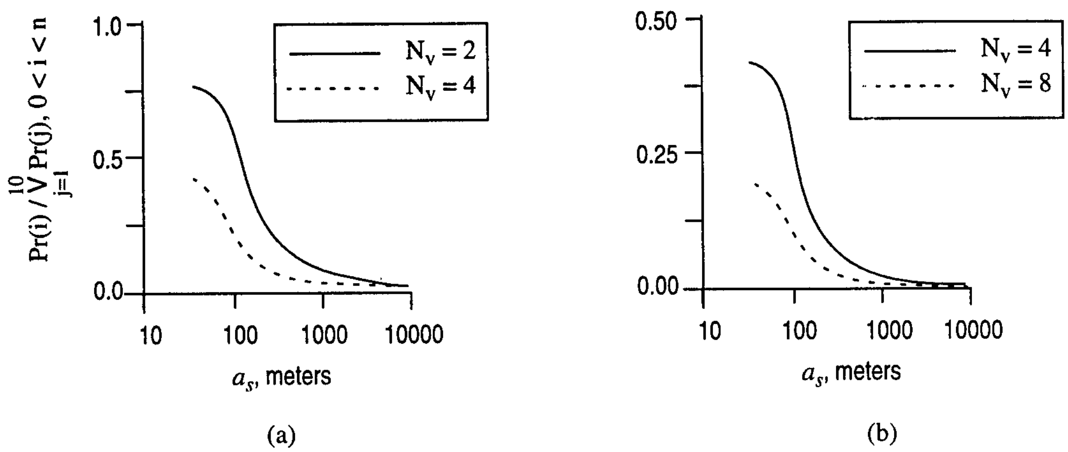


Figure 14. Variation of $Pr(N_v)$ with sensor altitude a_s, for $N_v = n/3$ and $N_v = 2n/3$, where (a) n = 6 and (b) n = 12.

Note that Figure 14 also portrays the effect of number of views. For example, in Figure 14b, where n = 12, the probabilities for $N_v = n/3 = 4$ and $N_v = 2n/3 = 8$ are approximately half of those in Figure 14a where $N_v = 2$ and $N_v = 4$. This scaling of N_v and $Pr(N_v)$ with respect to n appears to occur throughout our simulation runs, albeit with more discrepancies between probabilities associated with specific instances of N_v as a function of n. Hence, we are currently investigating ways to scale $Pr(N_v)$ with respect to n. For example, we have found that it is useful to measure the difference between probabilities obtained during simulation runs having small samples (e.g., small n), with the following measure of scaled probability:

$$M_{pn} = n \cdot Pr(l) \quad , \text{ where } \quad N_v = l \cdot n \quad \text{varies from one to n.}$$

Note that M_{pn} can be plotted against n to determine when the number of views is sufficient to support accurate probability estimation. For example, Figure 15 illustrates the difference in M_{pn} between successive settings of N_v, based on the simulation configuration from which Figure 13 was generated. Note that this difference measure, denoted by ΔM_{pn}, stabilizes in this case after approximately ten views. We are currently investigating conditions for the stability of ΔM_{pn} as a function of computational precision in other simulation scenarios.

However, the graph of Figure 14 can be misleading, since it foes not account for the ability to see *obliquely* under trees or overhanging cover. As shown in Figure 8b, it is naively assumed that if sensor altitude approaches cover altitude, then a downlooking sensor can view objects much farther away from a cover aperture intersected by the sensor's optical axis. Unfortunately, this is not the case in practice (as noted previously) because tree trunks, brush, and other types of physical clutter interfere with the sensor's oblique view. This phenomena increasingly becomes a source of occlusion as the look angle for a given pixel in the sensor's focal plane increases. This leads us to consider a simple modification to the model of Section 4.7, by which one can simulate the effect of three-dimensional cover.

Algorithm Enhancement. We have found that simulation of cover as a stack of L 2-D cover masks is sufficiently effective for the purposes of this model, provided that $L \geq 5$ is employed. The constraint on L becomes evident when one considers that a tree with spherical or ellipsoidal shape will have a bottom, equator, and top, with one slice required between the bottom and equator, and one slice between the equator and top, to portray the tree's cross section as a polygon approximating a round object, as shown in Figure 16a-c.

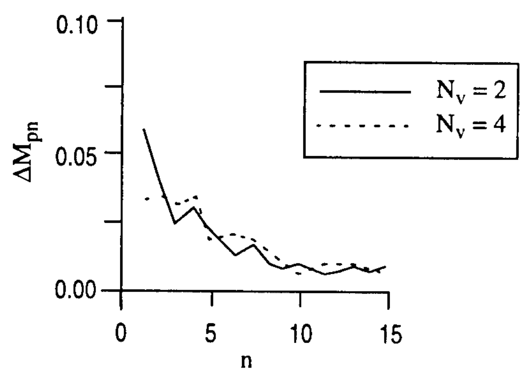


Figure 15. Graph of ΔM_{pn} as a function of n ranging from one to 15 views, based on the simulation configuration employed in the generation of Figure 13.

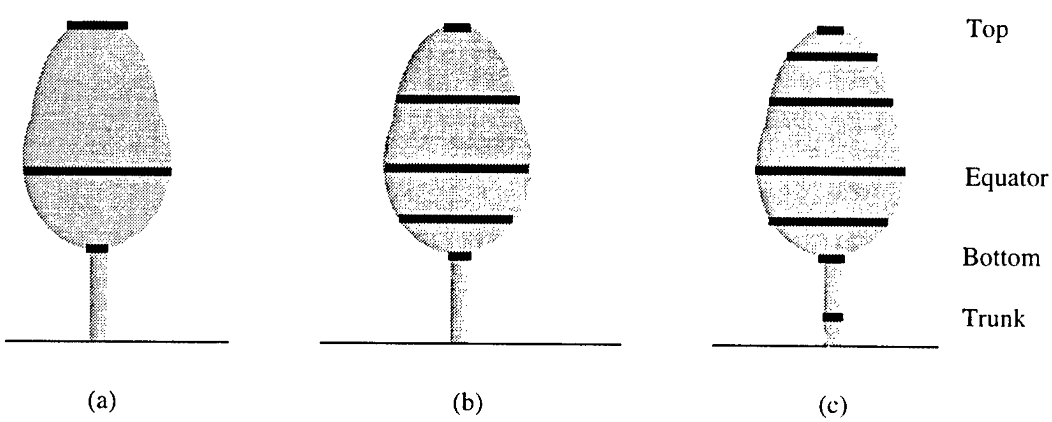


Figure 16. Approximation of 3-D arboreal cover cross-section with (a) three, (b) five, and (c) seven 2-D masks.

The enhancement to the simulation algorithm required to compute the effect of 3-D cover is quite simple, and is based on the observation that *any downward projection through any 3-D cover object represented as a stack of 2-D masks must penetrate all layers of cover if the projection intersects the target plane.* In practice, this means that, given m discrete two-dimensional cover masks $\mathbf{c}_\lambda \in [0, 1]^{\mathbf{Y}}$, where the stack index $\lambda = 1..m$, if a ray traced through the m masks intersects the λ-th mask at $(i_\lambda, j_\lambda) \in \mathbf{Y}$, then the transmittance t associated with that ray is given by

$$t = \prod_{\lambda=1}^{m} \mathbf{c}_\lambda(i_\lambda, j_\lambda) \ , \qquad \text{(XII)}$$

which follows from optics theory.

Hence, instead of tracing a ray from a sensor position **p** to a point in the target plane, we perform the ray tracing operation L times, to establish the set of coordinates

$$C(i, j) = \bigcup_{\lambda=1}^{m} \{(i_\lambda, j_\lambda)\}, (i, j) \in \mathbf{Y} \ .$$

Then one performs the multiplication shown in Equation (XII) to obtain the transmittance associated with the target pixel $\mathbf{d}(x_t,y_t)$, whose point $(x_t,y_t) = \mathbf{a}(i,j)$ is colinear with **p** and the points in $C(i,j)$, within the limits of colinearity of digital lines, given the spatial quantization error inherent in **X** and **Y**.

It is of further interest to note that the specification of a cover mask over the range space [0,1] permits the modelling of translucent cover (e.g., fog or haze), as well as the statistical characterization of cover transparency due to partial visibility within the subtense of one pixel of the cover mask (at IFOV $(\Delta x_c, \Delta y_c)$ physical units).

Example. Coniferous tree branches that are opaque but occupy only *k* percent of a cover pixel's subtense at the cover plane could be characterized by a transmittance of $k / 100$ at that pixel location.

5.2. Analysis of Cepstral-based Coregistration

Yeshurun and Schwartz [Yes89} describe a method of cepstral-based coregistration that we have employed in the experiments described in Section 6.1. In particular, given a stereo pair of MxN-pixel images, the stereo image formed as $\mathbf{c} = (\mathbf{a},\mathbf{b})$ is transformed via application of the cepstrum to yield the disparity image **d**, as follows:

$$\mathbf{d} = \mathcal{F}(\log[|\mathcal{F}(\mathbf{a}) + 1|]) \quad ,$$

where $\mathcal{F}$ denotes the Fourier transformation. The difference between the coordinates of the origin and largest peak of **d** yields the stereo disparity. This method is relatively impervious to additive noise, as shown in Figure 17a, which sensitivity analysis results incurred in correlating two instances of Figure 13b, where one instance is corrupted with Gaussian noise of mean 0.1 on a scale of [0,1] and standard deviation 0.02. Figure 17b (17c) illustrates correlation error incurred by rotating (scaling) the target over a range of [-15,15] degrees (scale factor range of [0.7,1.3]). Note that the results of Figure 17c for scale factor values less than 0.8 or greater than 1.2 are not shown, due to the large errors encountered at the extrema of the scaling range.

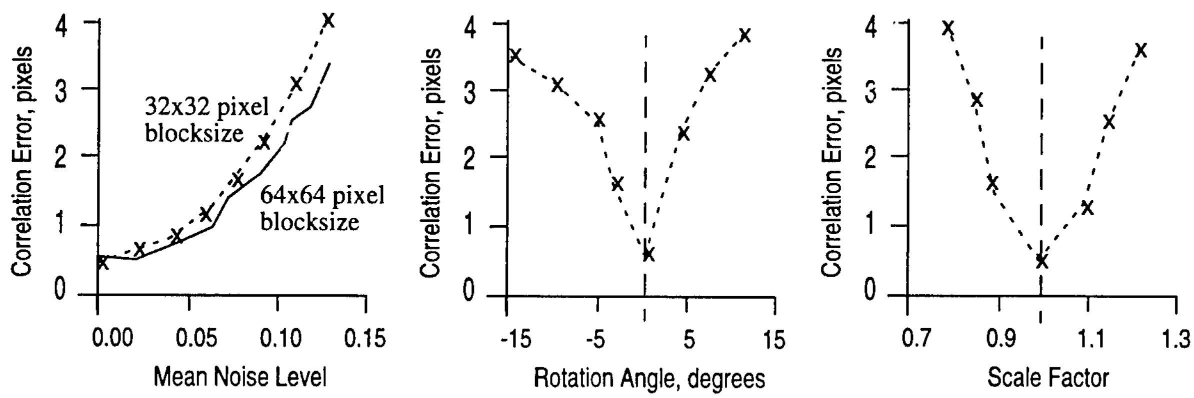


Figure 17. Correlation error for Yeshurun and Schwartz' cepstral coregistration of the pattern represented by Figure 13b under variable (a) noise, (b) rotation angle, and (c) scale factor, all other parameters being held constant.

The preceding analysis is presented in lieu of a detailed theoretical treatment, which is given in [Yes89] and [Smi96]. A key problem associated with analysis of the cepstral-based coregistration method is the analysis of error in the Fourier transform, whose implementations (and associated computational error) vary widely. This variation in computational error is also dependent on the configuration of the patterns being correlated and the size of the sample block. For example, in Figure 17, we employed a 1024x1024-pixel image, and varied the sample block size from 16x16 pixels to 64x64 pixels, choosing a blocksize of 32 pixels square as optimal. Slightly different results were obtained with different sized blocks, as shown in Figure 17a. Note that the 64x64-pixel block contains more information (i.e., higher spatial frequencies) than the smaller blocks, and thus produces less correlation error in the presence of noise. However, the computational cost associated with the 64x64-pixel block was insufficient to allow fast implementation and testing for the required number of simulation runs.

In Figures 17b and 17c, it is shown that the correlation error response is asymmetric with respect to the independent variable. This is due primarily to the asymmetry of the test pattern, and to local periodicity within the pattern (e.g., fine structure introduced by simulation model artifacts, since removed). In practice, such response asymmetry would result when correlating realistic targets in one view with the same targets in another view. This effect was observed when correlating simulated images of submerged targets, as discussed in Section 6.1.

Future work in the analysis of cepstral-based correlation involves (a) analyzing error propagation in different implementations of the Fourier transform, (b) testing Yeshurun and Schwartz' algorithm using the aforementioned Fourier transform implementations, as well as (c) implementing and testing Smith and Nandhakumar's enhancement [Smi96] of Yeshurun and Schwartz' algorithm using the several different z-transform implementations. We expect that this would provide a more extensive database of test results against which to estimate performance of cepstral-based coregistration in practice.

5.3. Analysis of Relative Geometric Reconstruction Method

The change in the 2-D projection of a moving 3-D object or camera yields important information for 3-D object

MODEL-BASED ASSESSMENT OF CAMPAIGN PLAN PERFORMANCE UNDER UNCERTAINTY

Nong Ye
Associate Professor
Department of Industrial & Management Systems Engineering

Vuong Nguyen
Research Assistant
Department of Industrial & Management Systems Engineering

George Runger
Associate Professor
Department of Industrial & Management Systems Engineering

Arizona State University
Tempe, AZ 85287-5906

Final Report for:
Summer Research Extension Program

Sponsor by:
Air Force of Scientific Research
Bolling Air Force Base, DC

March 1999

MODEL-BASED ASSESSMENT OF CAMPAIGN PLAN PERFORMANCE UNDER UNCERTAINTY

Nong Ye
Associate Professor
Department of Industrial & Management Systems Engineering

Vuong Nguyen
Research Assistant
Department of Industrial & Management Systems Engineering

George Runger
Associate Professor
Department of Industrial & Management Systems Engineering

Abstract

The command and control (C^2) of an air campaign involves different processes: planning, scheduling, execution, monitoring, assessment, and re-planning. The monitoring process updates the situation of the air campaign and detects changes in the campaign by processing intelligent information. The assessment process assesses those changes by associating the current situation with the objective of the campaign. Together, these two processes provide appropriate feedback to operation commander for making decisions regarding to the fate of the air campaign. This project proposes methods to assess a campaign by fusing intelligent information. There are many methods for data fusion, three methodologies discussed in this project including conventional method, Dempster-Shafer method, and maximal method. The proposed methods mainly focus on how to transform intelligent information into a meaningful picture of battlefield situation. An air campaign may have several operations, some operations are parallel and some operations are sequential. Given the probabilities of delay or success of all or some operations, the proposed methods attempt to draw the conclusion about the probability of success of the campaign. In addition, the proposed methods determine the total time of delay if there is delay. The conventional method simply draws a conclusion based on largest probability of all parallel operations. The Dempster-Shafter theory fuses multiple probabilities to give conclusion about the probability of delay of the campaign. The maximal method is a classical statistical method in which parallel operations are combined into an operation so that all tasks are sequential in order to applying normal distribution property.

MODEL-BASED ASSESSMENT OF CAMPAIGN PLAN
PERFORMANCE UNDER UNCERTAINTY

Nong Ye, Vuong Nguyen, George Runger

1. Introduction

The command and control (C^2) of an air campaign involve different processes (planning, scheduling, execution, monitoring, assessment, and re-planning) in different functional areas (force application, force enhancement, and force support) at different levels (from strategic to tactical) [1-3]. Currently, C^2 operations suffer from a number of shortfalls [3]. First, the planning of an air campaign is cyclic by a daily or weekly timeline, which results in a slow and inefficient response to changes in campaign objectives and battlefield situations during plan execution. Second, there is a lack of integration between C^2 processes. Third, the C^2 processes are human intensive. It requires days for planning, scheduling, monitoring, assessment, and re-planning. Therefore, the integration and automation of C^2 operations is highly desirable to move C^2 operations from schedule-driven to situation-driven continuous operation. Integration and automation improve the responsiveness to dynamically changing campaign objectives and battlefield situations from days to hours or minutes.

Through the Summer Faculty Research Program, an integration infrastructure was established to support continuous C^2 operations based on a process-engineering schema for the control and analysis of dynamic process systems in industrial environment [4]. To demonstrate the feasibility of this processing engineering approach to the integration of C^2 operations, a process model was defined using an air campaign scenario. Research work on the automation of C^2 operations is proposed here to provide a more comprehensive vision of the process engineering approach to continuos C^2 operations.

To support integration and automation of C^2 operations, appropriate methods need to be identified for all processes. However, due to time limitation, only data fusion methods for the monitoring and assessment processes are addressed in this proposal. Monitoring and assessment processes can be considered as an integrated process, one updates the situation of the air campaign, detect changes of campaign plans using Intelligence, Surveillance, and Reconnaissance (ISR) information. The other assesses the changes by associating the current situation with current objectives of the campaign plan. Together, they provide feedback to operation commander for making decisions regarding the current situation of the campaign.

As stated above, an air campaign plan can be represented as a dynamic system in which the performance of one task affects the performance of subsequent tasks. A change to a task may associate with the preconditions or the execution of the task, which in turn causes a change in the start time, end time, and duration of the task. More severely, it may change the whole situation of the campaign plan. We need to assess the performance of an air campaign based on ISR information regularly to determine whether the campaign plan should be continued, re-planned, or terminated. In other words, periodic campaign assessment improves the responsiveness to dynamic battlefield situations.

In our work, there are three methods will be utilized to fuse ISR information and assess the campaign plan including conventional method, Dempster-Shafer method, and maximal method. Section 2 introduces an air campaign scenario and the process model of the air campaign. Section 3 defines the meaning of confident interval and measure of merits (MoM), which will be used extensively to evaluate the performance of proposed techniques. Section 4 proposes methods to fuse ISR information into a meaningful picture of battlefield situation. Section 5 tests and analyzes the performance of those techniques.

2. An Process Model of an Air Campaign

The process-engineering schema has traditionally been used to model industrial system (e.g. discrete part or continuous process manufacturing systems) for the planing, control, and analysis purposes. An industrial process system is modeled at four levels of abstraction: objective, conceptual, functional, and physical. At the objective level, the goals of the systems are stated. At the conceptual level, the goals of the systems are transformed into operational definitions of system behavior in terms of time phased and sequenced tasks. At the functional level, tasks at the conceptual level are transformed into tasks, which are performed by functional units of the system. At the physical level, tasks at the functional level are implemented by events that are executed by physical units of the system. From the objective level to the physical level, the focus moves from the global, abstract view of the system to the local, specific view of system components and their relationships.

Similarly, an air campaign can be modeled at four levels, which are objective level, conceptual level, functional level, and physical level. At the objective level, campaign objectives are defined including start states, intermediate states, and desired end states of the air campaign. In addition, the objective level specifies constraints if there are any. In the campaign scenario under consideration, the objectives of the 'blue' force are to rescue a delegation in a US Embassy under the control of a 'red' force, and to secure an airport under the control of the 'red' force (Figure 1). These two objectives must be simultaneously accomplished [4].

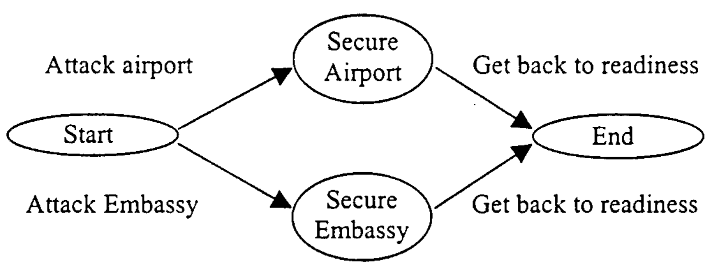


Fig. 1. The 'blue' force needs to rescue a delegation in a US Embassy and secures the airport under the control of the 'red' force.

At the conceptual level, campaign objectives are expended into phased and sequenced states. Conceptual states reflect situations of the air campaign system at critical moments of the campaign plan. Integrated joint force tasks are not segregated into different functional areas such as force application (FA), force enhancement (FE), and force support. However, the blue force is refined into smaller groups, which can be US force and allied force, and their tasks.

The strategic plan of the air campaign at the conceptual level is supported by a tactical plan of the functional level. Tasks of the US force and the allied force are further decomposed into tasks of functional areas (FA, FE, and FS).

At the physical level, tactical plan at the functional level is further refined into a tactical plan dealing with execution of each functional component. Components and tasks are physically identified. They are mapped in their raw forms such as images of aircraft, roads, bridges, etc.

In our work, we focus on an air campaign plan at the physical level. The air campaign process is represented by a sequence of tasks in which the delay of a task will result in the delay of the starting time of subsequent tasks. However, the delay of a task does not affect the duration of other tasks dues to our task independent assumption. Figure 2 shows the tasks of US force and the expected duration (μ) of tasks. In addition, the standard deviation (σ) of a task is specified to be 1/8 of its duration.

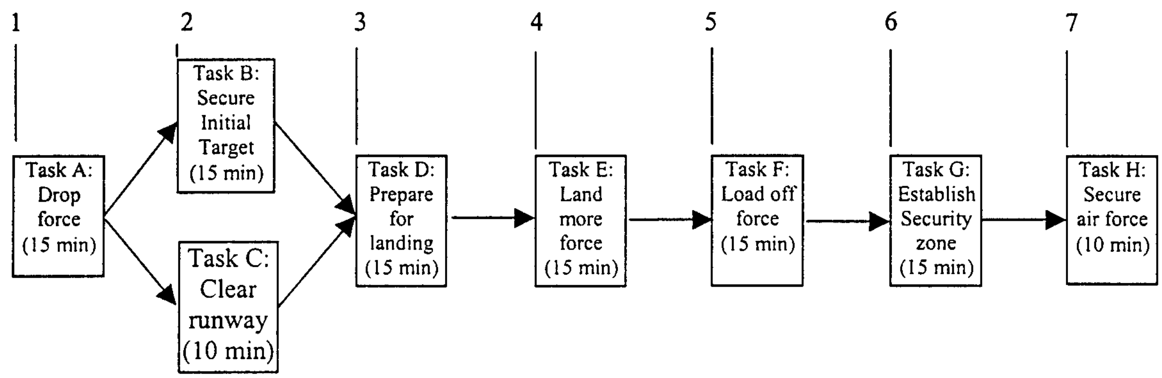


Fig. 2. Activities of US force at the physical level. Numbers on the top of boxes indicate the decision time when ISR information arrives. Number inside the boxes indicates the expected duration of tasks.

3. The Definition of MoM and Confidence Value

A measure of merits (MoM) of a task is a composite index that indicates the degree of satisfaction of achieving that task. A negative MoM indicates that the task is accomplished earlier than the expectation. A positive MoM indicates a delay in accomplishing the task. A MoM value of zero indicates that the task is accomplished right on time. However, providing the MoM index without a confidence level is meaningless. A confidence level indicates the degree of belief in a delay of accomplishing task. For instance, at the decision time 1, ISR information report that task A may be delayed for 4 minutes with 80 percent confidence (MoM = 4, confidence = 80%). Task 1 may be delayed for –1 minute with 90 percent confidence (MoM = -1, confidence = 90%). Based on this information, proposed methods make inference and recommendation about the whole campaign plan to determine whether the campaign needs to be modified. As the time goes on, ISR information of more tasks are available, the accuracy of campaign assessment will increase. If the ISR information reports that task A cannot be completed at all due to some reasons (MoM = ∞ at 100% confidence level), then the air campaign plan should be terminated or re-planned.

4. Information Fusion Techniques

There are various types of technique dealing with reasoning under uncertainty, including statistical methods (e.g., hypothesis testing), probabilistic methods (e.g., Bayes' rule, Bayesian networks), certainty factor, Dempster-Shafer theory, and fuzzy logic [5-7]. Many studies have proposed a stand-alone technique to deal with uncertainty situation. We develop hybrid methods those utilize both classical technique and Dempster-Shafer for information fusion. The proposed methods address the problem in two steps. The first step will be the interpretation of ISR information. There are three possible interpretations for raw ISR information, including discrete distribution, conservative continuous distribution and optimistic continuous distribution. The second step is to apply the proposed methods to the processing ISR information to assess the battlefield situation.

Consider the following example. ISR reports that task A (MoM = 1, confidence = 95%), task B (MoM = 1, confidence = 98%), task C (MoM = -5, confidence = 90%), and other tasks are unknown. Since there is no information for other tasks, their duration assume remaining the same. We will use this example to demonstrate the proposed methods for all three interpretations.

4.1. Discrete Distribution

If the duration of tasks follow discrete distribution, the new duration of tasks A, B, and C are updated for the given example as follows.

$$A=\begin{cases}10 & (5\%)\\ 11 & (95\%)\end{cases} \qquad B=\begin{cases}15 & (2\%)\\ 16 & (98\%)\end{cases} \qquad C=\begin{cases}10 & (10\%)\\ 5 & (90\%)\end{cases}$$

The duration of other tasks remains the same as given in Figure 2. Figure 3 shows the discrete distribution of task A. Tasks B and C can be represented in the same manner.

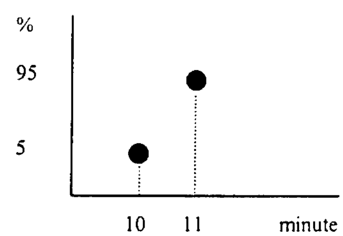


Fig. 3. Discrete distribution: Task A will delay 1 minute at 95 percent confidence, and will be on time at 5 percent confidence level [(11, 0.95), (10, 0.05)].

The process model given in Figure 2 can be decomposed into two paths: path 1 includes tasks A, B, D, E, F, G, and H, path 2 includes tasks A, C, D, E, F, G, and H. In current example, we know two tasks for both path 1 (tasks A and task B) and path 2 (task A and task C), therefore, we have four combinations for each path. Among 4 combinations, whichever has total duration greater than 90 (the expected duration of the campaign plan), their probability will be summed up to calculate the probability of delay. The total of operation duration and probability of delay for the two paths are calculated by following equation

$$\mu_T=\sum_i \mu_i \qquad P_T=\sum P(D_i) \qquad where \quad P=\prod_j P_j \tag{1}$$

Note that μ_i is the duration of tasks. P_T is the probability of delay of the operation, and P is the probability of delay of given combination.

From equation (1), the total operation time and confidence level of path 1 are

Combination 1: task A is NOT delayed and task 1 is NOT delayed.

$$\mu_T = \boldsymbol{10} + \boldsymbol{15} + 10 + 15 + 15 + 15 + 10 = 90 \qquad (\leq 90)$$

$$P = 0.05 * 0.02 = 0.001$$

Combination 2: task A is NOT delayed but task 1 is delayed

$$\mu_T = \boldsymbol{10} + \boldsymbol{16} + 10 + 15 + 15 + 15 + 10 = 91 \qquad \mathbf{(>90)}$$

$$P = 0.05 * 0.98 = 0.049$$

Combination 3: task A is delayed but task 1 is NOT delayed

$$\mu_T = \boldsymbol{11} + \boldsymbol{15} + 10 + 15 + 15 + 15 + 10 = 91 \qquad \mathbf{(>90)}$$

$$P = 0.95 * 0.02 = 0.019$$

Combination 4: task A is delayed and task 1 is delayed.

$$\mu_T = \boldsymbol{11} + \boldsymbol{16} + 10 + 15 + 15 + 15 + 10 = 92 \qquad \mathbf{(>90)}$$

$$P = 0.95 * 0.98 = 0.931$$

Combinations 2, 3, and 4 have μ_T greater than 90; therefore, are summed up to calculate the MoM and the probability of delay of path 1 are:

$$MoM = (91-90)*0.049 + (91-90)*0.019+(92-90)*0.931=1.93$$

$$P = \Sigma P\ (D_i) = 0.049 + 0.019 + 0.931 = 0.999$$

For path 1, its MoM and confidence level are calculated in similar manner, which yields following results

$$MoM = (85-90)*0.05+(80-90)*0.045+(86-90)*0.095+(81-90)*0.855=-8.775$$

$$P = \Sigma P\ (D_i) = 0\%$$

Having the probability of delay of two paths, we can calculate the probability of delay for the whole campaign plan by using the proposed methods.

Conventional Method: Conventionally we just select the maximum probability of delay of two paths. In this example, we conclude that the campaign will be delayed for 1.93 minutes at 99.9% confidence level.

Dempster-Shafer Method: Equation 2 is a Dempster's rule of combination, which can be used to combine the probability of delay of the operations.

$$m_1 \oplus m_2(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) m_2(Y)}{1-\kappa} \qquad \mathbf{(2)}$$

Table 1. Combining probability of delay of US operation and allied operation

	$m_2(\{D\})=0$	$m_2(\Theta)=1$
$m_1(\{D\})=0.999$	$m_1 \oplus m_2(\{D\})=0$	$m_1 \oplus m_2(\{D\})=0.999$
$m_1(\Theta)=0.001$	$m_1 \oplus m_2(\{D\})=0$	$m_1 \oplus m_2(\Theta)=0.001$

Where $m_1(\{D\})$ is the mass of evidence of delay of path 1

$m_1(\Theta)$ is the degree of disbelief in evidence of delay of path 1

$m_2(\{D\})$ is the mass of evidence of delay of path 2

$m_2(\Theta)$ is the degree of disbelief in evidence of delay of path 2

Using Dempster_Shafer's rule of combination, the probability of delay of the campaign is

$$m_1 \oplus m_2(\{D\}) = 0.999 + 0 + 0$$
$$= 0.999$$

We can conclude that the campaign will be delayed for 1.93 minutes at 99.9 percent confidence [7, 8, 9, 10]

4.2. Continuous Distribution: Optimistic Interpretation.

In optimistic interpretation, task A will be accomplished between 10 minutes and 11 minutes at 95 percent confidence level [P ($10 \le A \le 11$) = 0.95] (Figure 4).

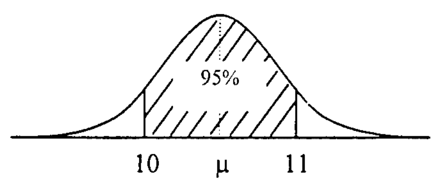


Fig. 4. Optimistic interpretation for task A

New mean (μ) and standard deviation (σ) of tasks need to be re-calculated. There are several ways to calculate new μ. In our work, new μ will be calculated as the average of two given values. For task A, μ_A is 10.5, which is the middle of 10 and 11. Since we know μ_A and probability (P=0.95), we can calculate σ_A.

$$Z = \frac{X-\mu}{\sigma} \Rightarrow \sigma = \frac{X-\mu}{Z} = \frac{11-10.5}{Z(P=0.975)} = 0.2551$$

In the same manner, the new μ of tasks B and C are 15.5, and 7.5, respectively. The new σ of tasks A is 0.2149 and task B is 1.5199. Since there is no new information of other tasks, their μ and σ remain the same.

The total of operation duration and its standard deviation of distribution for path 1 and path 2 are calculated based on the probability of normal distribution [11]

$$\mu_T = \sum_i \mu_i \qquad \sigma_T = \sqrt{\sum_i \sigma_i^2} \tag{3}$$

From Equation 3, new μ_T and σ_T of path 1 are

$$\mu_T = \mu_A + \mu_B + \mu_D + \mu_E + \mu_F + \mu_g + \mu_H = 10.5 + 15.5 + 10 + 15 + 15 + 15 + 10 = 91$$
$$\sigma_T = \sqrt{0.255 + 0.215 + 1.563 + 3.516 + 3.516 + 3.516 + 1.563} = 3.7126$$

Similarly, new μ_T and σ_T of path 2 are

$$\mu_T = \mu_A + \mu_C + \mu_D + \mu_E + \mu_F + \mu_G + \mu_H = 10.5 + 7.5 + 10 + 15 + 15 + 15 + 10 = 83$$
$$\sigma_T = \sqrt{0.255 + 1.520 + 1.563 + 3.516 + 3.516 + 3.516 + 1.563} = 4.006$$

Given distribution of the operations, we can calculate the probability that the campaign will be delayed. From Figure 5, the probability of delay for path 1, path 2 is 0.606, 0.04, respectively.

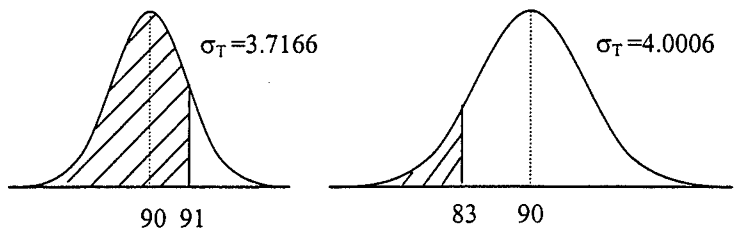


Fig. 5. Distribution of path 1 and path 2, respectively

Given the probabilities of delay of the two paths, we can calculate the probability of delay for the whole campaign plan by using the proposed methods.

Conventional Method: Perform calculation as in discrete distribution, we conclude that the campaign plan will be delayed for 1 minute at 60.6 percent confidence.

Dempster-Shafer Method: Perform calculation as in discrete distribution, we conclude that the whole campaign plan will be delayed for 1 minute at 62.2 percent confidence.

Maximal Method: Instead of decomposing the campaign (Figure 2) into two paths, the maximal method combines parallel tasks into single task and applies the property of the normal distribution to inference the distribution of the whole campaign. In this scenario, task B and C are parallel, the maximal method select the maximum between these two tasks, called task Y (Figure 6) by using equation 4.

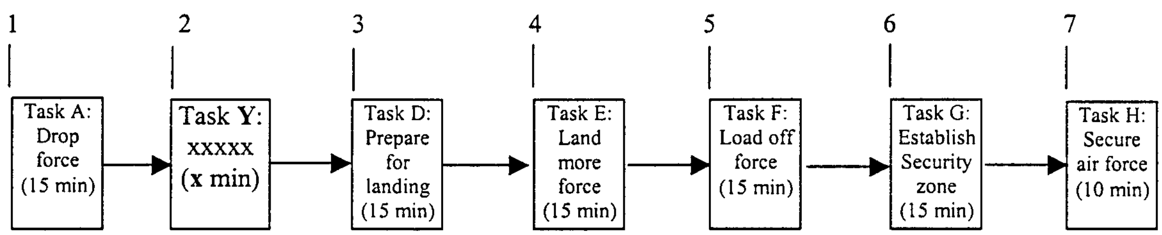


Fig. 6. Using maximal method to combine two parallel tasks into single task

$$F(y) = P\big(z = \max(B,C) \le y\big) = P(B \le y) * P(C \le y) \tag{4}$$

$$f(y) = \frac{\partial P}{\partial y} = P(B \le y)\frac{\partial P}{\partial y} + \frac{\partial P}{\partial y}P(C \le y)$$

$$f(y) = \Phi\left(\frac{y-\mu_B}{\sigma_B}\right) * f_v(y) + \Phi\left(\frac{y-\mu_C}{\sigma_C}\right) * f_C(y)$$

Then task Y will be combined with task X, which is the sum of serial tasks (X = A + C + D+ E + F + G + H) using Equation 5.

$$P(X+Y \le 90) = EP(X+Y \le 90 \mid Y = y)$$
$$P(X+Y \le 90) = E_y P(X \le C - y)$$
$$P(X+Y \le 90) = \int_{9.38}^{20.6} \Phi\left(\frac{90 - y - \mu_X}{\sigma_X}\right) * f(y)dy \qquad (5)$$

It is reasonable to assume the range of any task will be its mean plus 3 standard deviations. Therefore, the maximum and minimum value of task B will most likely be $\mu_B \pm 3\sigma_B$, which are 20.6 for maximum value and 9.38 for minimum value. These two values will be used as upper and lower bound for the integration in the equation

Equation 5 yields the probability of not delay, so the probability of delay will be 1 - P(X + Y ≤ 90). Using equation 4 and 5, the probability of delay of this example is 60.62 % and MoM is 1.93 minutes.

4.3. Continuous Distribution: Conservative Interpretation.

In a conservative interpretation, task A needs at least 11 minutes to be done at 95 percent confidence [P (A ≥ 11) = 0.95] (Figure 7). Other tasks can be interpreted in the same manner.

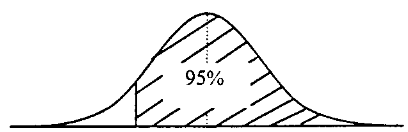


Fig. 7. Conservative interpretation for task A

Considering task A again, since we know its probability and its σ, we can calculate μ

$$Z = \frac{X-\mu}{\sigma} \Rightarrow \mu = X - \sigma * Z(P = 1 - 0.95) = 11 - \frac{1}{8} * 11 * (-1.6449) = 13.262$$

In similar manner, new μ of tasks 1 and 2 are 20.107 and 5.801, respectively. The σ of tasks A is 1.38, task 1 is 2, and task 2 is 0.63. μ and σ of other tasks remain the same.

How to obtain the μ_T and σ_T for the distributions of path 1 and path 2 is similar to the optimistic interpretation case. From Equation 3, the μ_T and σ_T of path 1 are 98.4 and 4.423, respectively. The μ_T and σ_T of path 2 are 84 and 3.994, respectively.

Similar to optimistic interpretation, given distribution of both paths, we can calculate the probability that the campaign will be delayed. The probabilities of delay of path 1 and path 2 are 0.9708, 0.0686, respectively (Figure 8). From these probabilities, the probability of the whole campaign plan can be obtained using proposed methods.

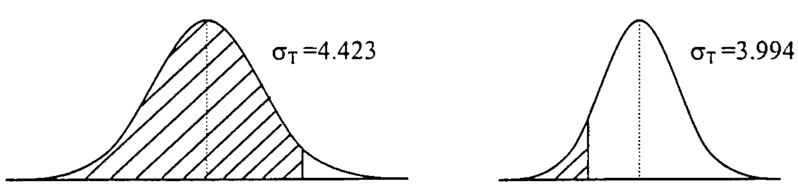


Fig. 8. Distribution of path 1 and path 2, respectively

Conventional Method: Similar to calculation in discrete distribution, we conclude that the whole campaign plan will be delayed for 8.4 minutes at 97.08 percent confidence.

Dempster-Shafer Method: Similar to calculation in discrete distribution, we conclude that the whole campaign plan will be delayed for 8.4 minutes at 97.28 percent confidence

Maximal Method: Similar to calculation in optimist interpretation, we conclude that the whole campaign plan will be delayed for 1.93 minutes (or 2 minutes) at 83.28 percent confidence

5. Testing and Results

We create a numbers of testing cases for the air campaign scenario in Figure 2 to exam the performance of the proposed techniques. Cases can be classified into different types based on the number of task that ISR reports ahead of time at each decision time. ISR can report the MoM and probability of delay of one task, two tasks, or all tasks ahead of time. In this project, three types of testing cases are generated. Type 1 is test case for reporting one task ahead of time. Similarly, type 2 and type 3 is test case for reporting two tasks and three tasks ahead of time, respectively. There are five scenarios for each type of test case. Appendixes 1, 2, and 3 are scenarios for reporting 1, 2, and 3 tasks ahead of time, respectively.

Table 2 shows several scenarios of type 1 test case. Consider scenario 1 at time 1 (T1), ISR only reports the MoM and the probability of delay for task A, which is MoM = 5 and confidence = 77 %. Information of other tasks is unknown. At T2, probability of task A is 100% because task A have already completed. From this, task A is accomplished 5 minutes earlier than expectation. Task B and C predicted to be completed earlier than expected by 4 minutes at 79 % confidence and 5 minute at 88 % confidence, respectively. As time is moving on, more tasks will be known with certainty, and so the campaign. For instance, at T7 the probability of delay for the whole scenario can be predicted without much variation since tasks A, B, C, D, E, F, and G are known with certainty, but task H.

5.1. Results

For each decision time of a testing case, we use the proposed methods to determine the MoM and the confidence value for the whole campaign scenario at that decision time. This pair of MoM and confidence values is then compared with the pair of MoM and confidence values obtained from the simulation that we assume represents the real situation of the plan execution. Three interpretations represent totally different populations; therefore, three simulations will be created for three corresponding proposed interpretations. Appendixes 4, 5, and 6 are results of type 1, type 2, and type 3 test cases for discrete distribution. Appendixes 7, 8, and 9 are results of type 1, type 2, and type 3 test case for optimist interpretation. Appendixes 10, 11, and 12 are results of type 1, type 2, and type 3 test case for conservative interpretation.

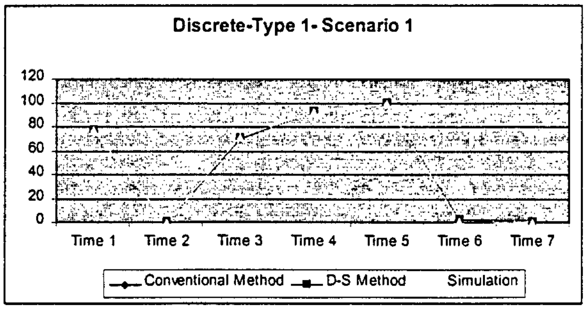
Table 2. Several test cases for type 1: reporting one task ahead of time

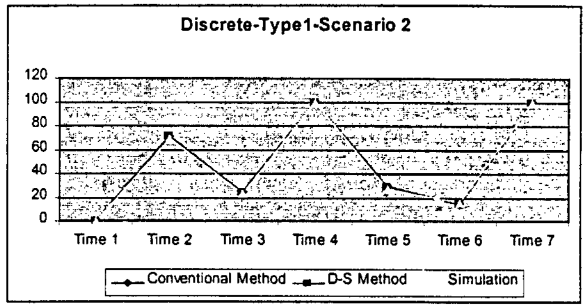
		SCENARIO 1		SCENARIO 2		SCENARIO 3		SCENARIO 4		SCENARIO 5	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	4	77	-2	93	1	71	2	79	1	78
	B										
	C										
Time 1	D										
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	-4	79	5	71	1	99	-2	99	-2	87
	C	-5	88	-2	99	-4	75	-3	73	5	86
Time 2	D										
	E										
	F										
	G										
	H										
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	100	2	100	-1	100	-3	100	-5	100
	C	1	100	-4	100	-3	100	-5	100	-2	100
Time 7	D	5	100	1	100	-4	100	-2	100	-4	100
	E	-2	100	-5	100	-1	100	-4	100	-4	100
	F	-1	100	5	100	5	100	-1	100	3	100
	G	-1	100	-1	100	3	100	5	100	0	100
	H	1	77	3	74	-1	94	4	84	-1	85

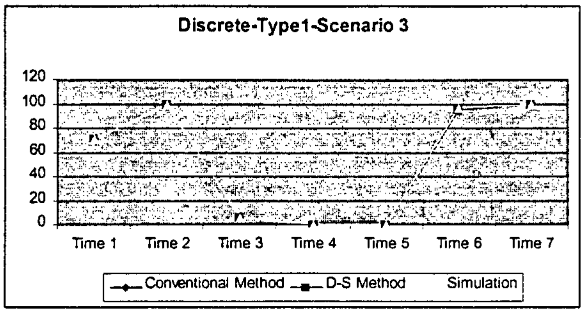
5.2. Performance Analysis

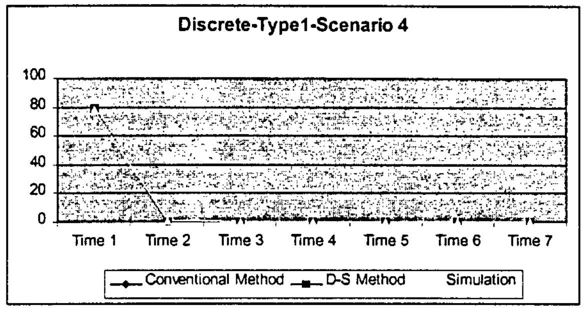
From the plots, the proposed methods have demonstrated theirs superior performance in determining the probability of delay of the campaign plan. Methods for determining of MoM need to be refined for more accuracy (see Appendixes 4-12). Among three interpretations, discrete interpretation has shown better results than the others. However, the discrete method requires more calculation than the others. Among proposed methods, the Dempster-Shafer and the conventional method have relatively shown theirs superior to the maximal method although the calculation of maximal method is more complex than of the conventional method and the Dempster-Shafer theory method. Therefore, the Dempster-Shafer and the conventional method are recommended for future use.

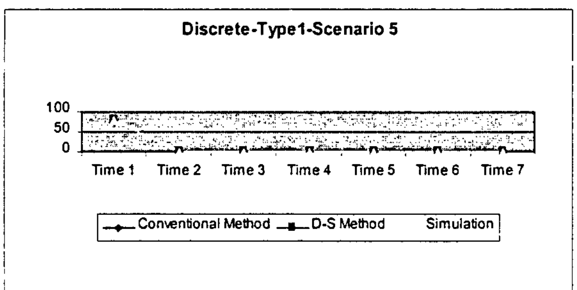
Graphical Comparison of Confidence Values of Proposed Methods and Simulation for Discrete Distribution

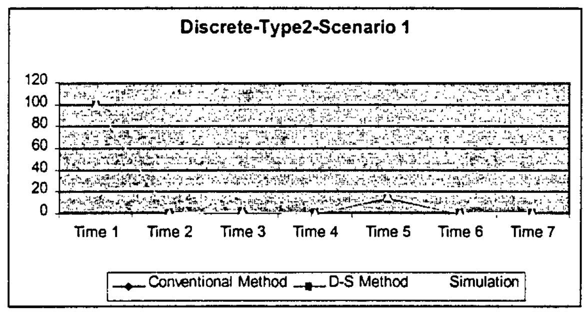


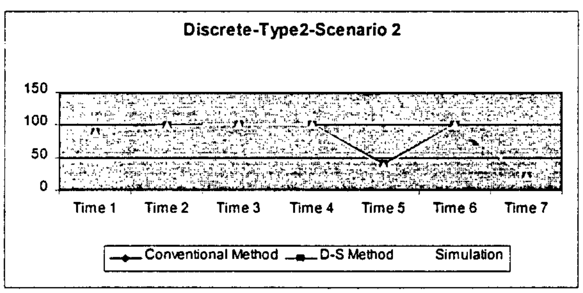


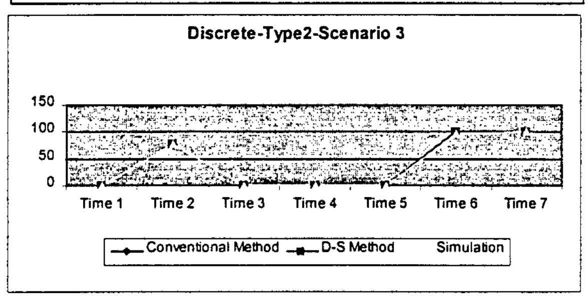


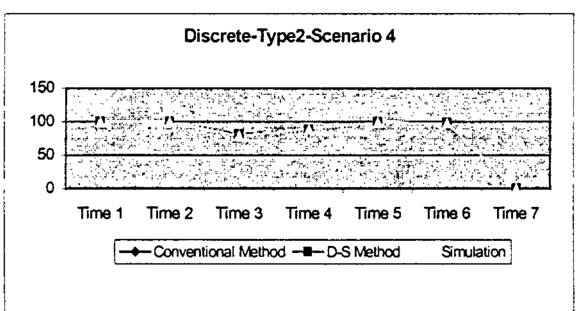


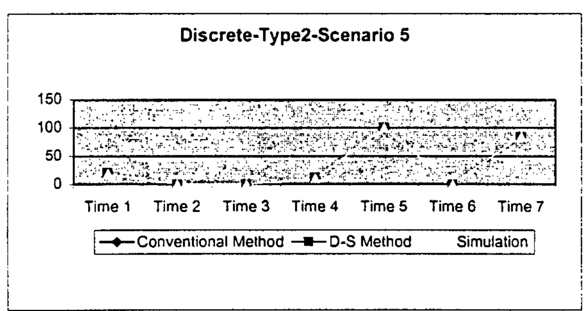


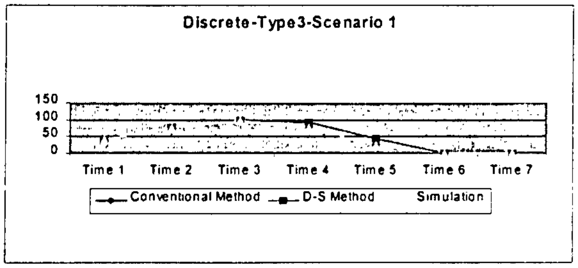


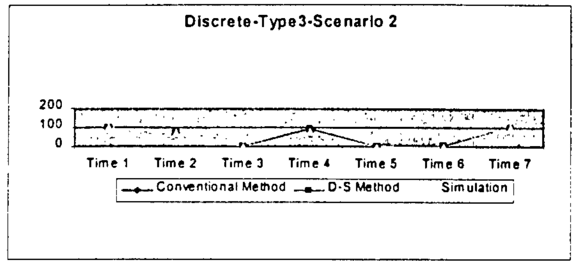


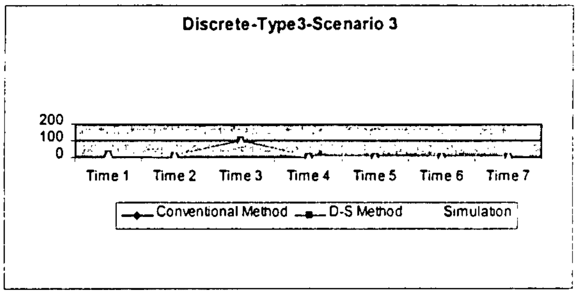


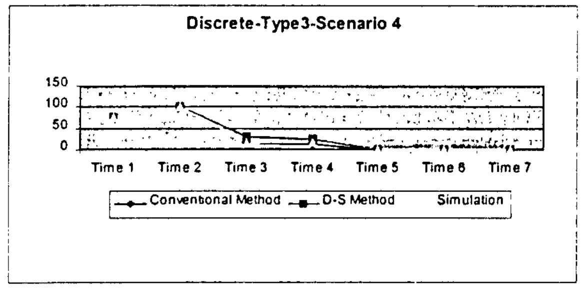


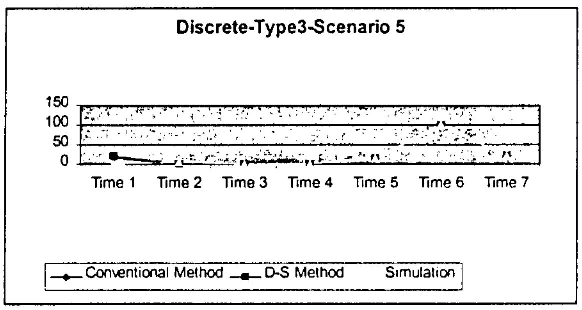




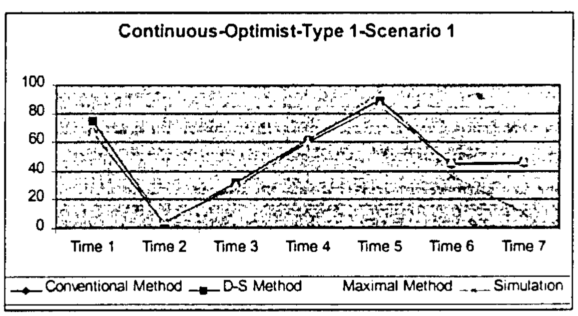


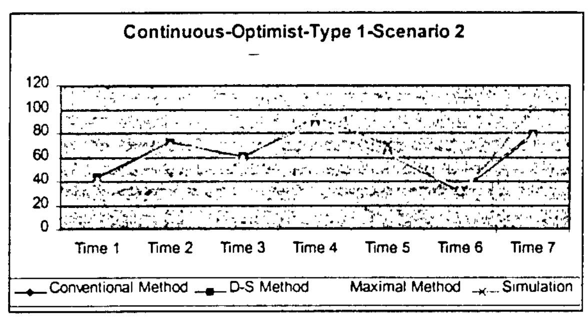


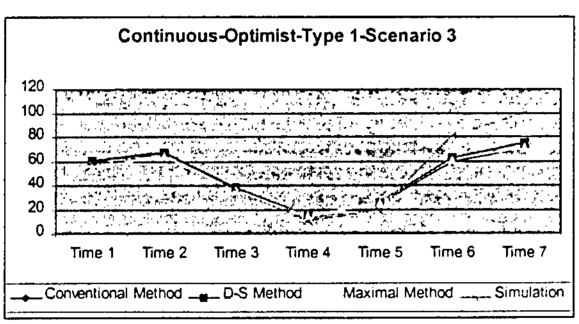


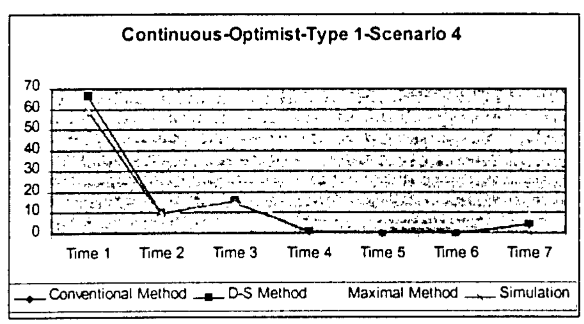


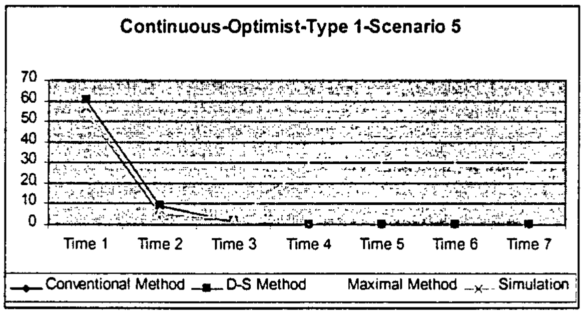
Graphical Comparison of Confidence Values of Proposed Methods and Simulation for Optimist Interpretation

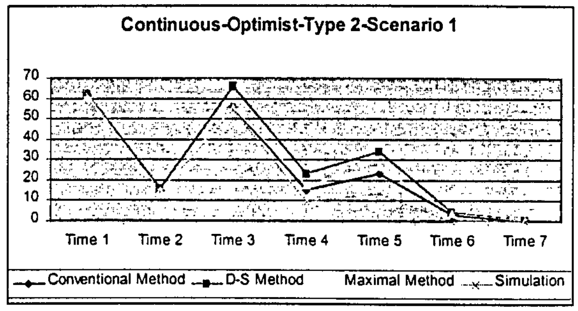


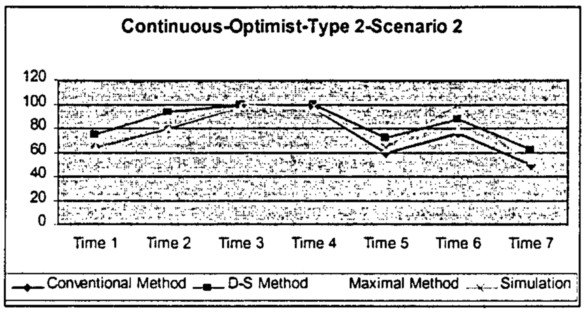


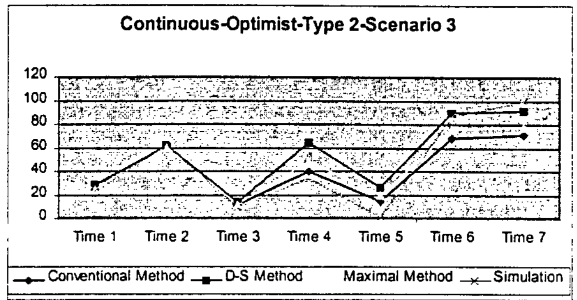


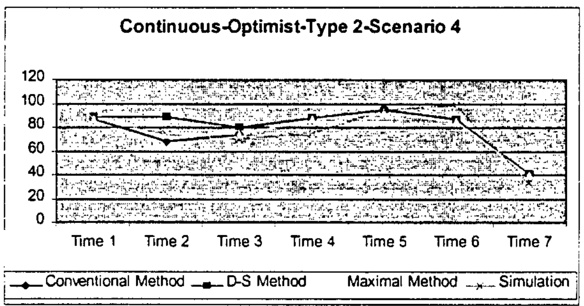


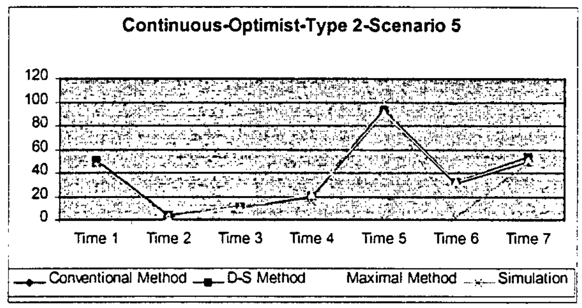

Continuous-Optimist-Type 1-Scenario 5
70
60
50
40
30
20
10
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

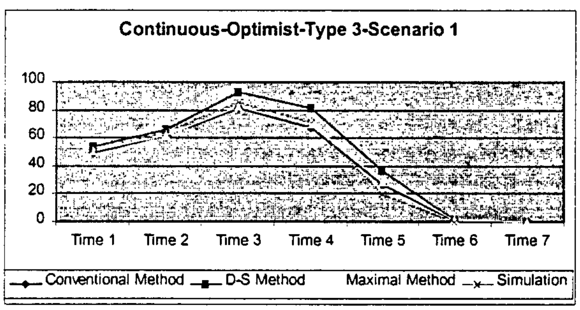

Continuous-Optimist-Type 2-Scenario 1
70
60
50
40
30
20
10
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

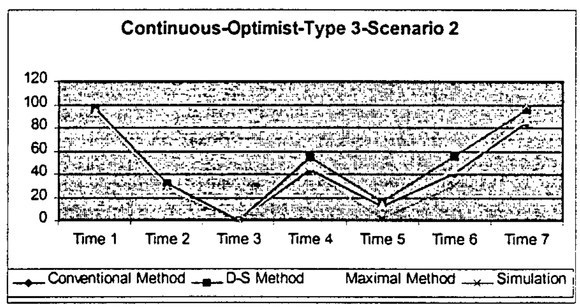

Continuous-Optimist-Type 2-Scenario 2
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

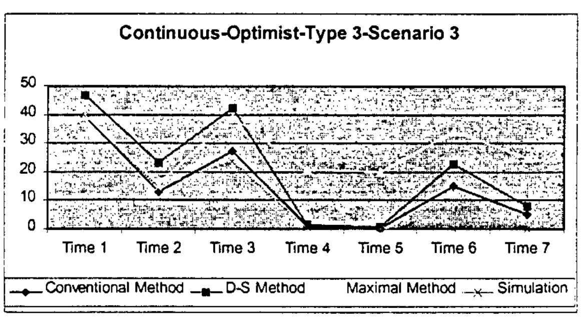

Continuous-Optimist-Type 2-Scenario 3
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

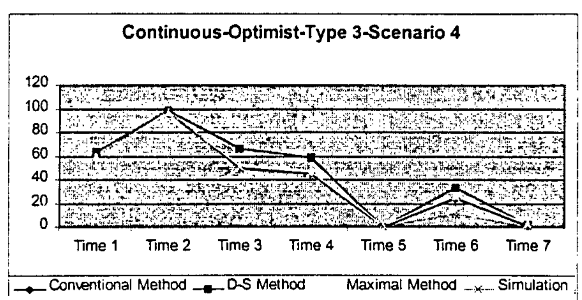

Continuous-Optimist-Type 2-Scenario 4
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

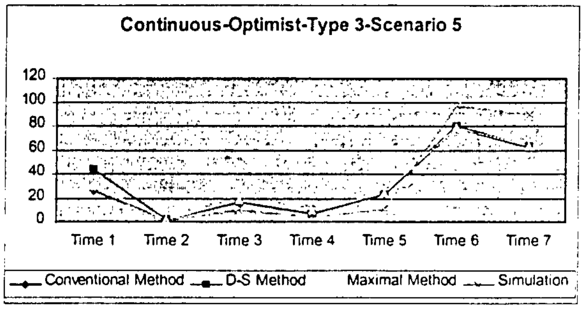

Continuous-Optimist-Type 2-Scenario 5
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation


Continuous-Optimist-Type 3-Scenario 1
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

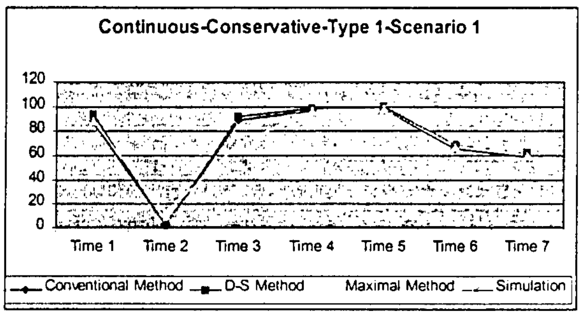

Continuous-Optimist-Type 3-Scenario 2
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

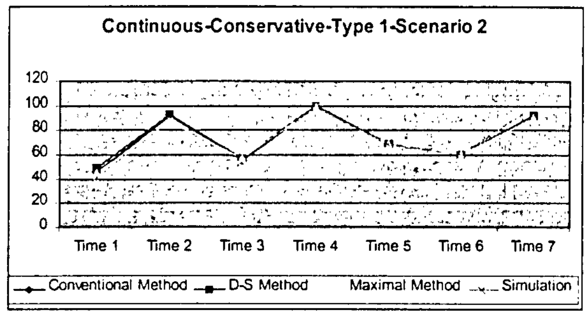

Continuous-Optimist-Type 3-Scenario 3
50
40
30
20
10
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

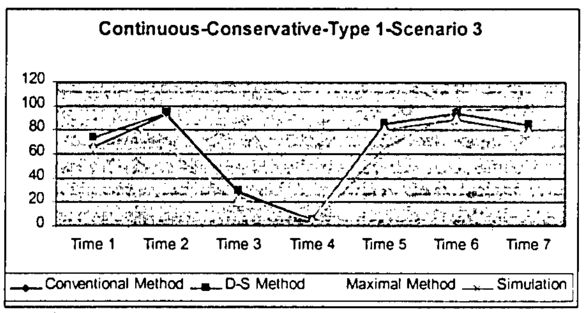

Continuous-Optimist-Type 3-Scenario 4
120
100
80
60
40
20
0
Time 1
Time 2
Time 3
Time 4
Time 5
Time 6
Time 7
Conventional Method
D-S Method
Maximal Method
Simulation

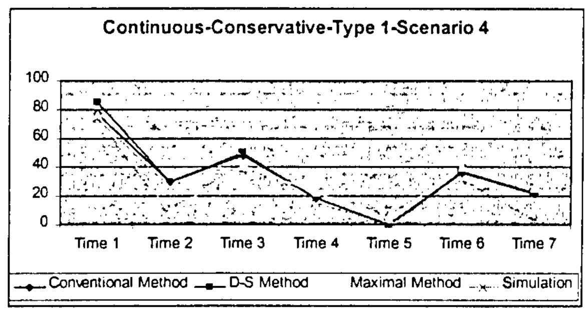


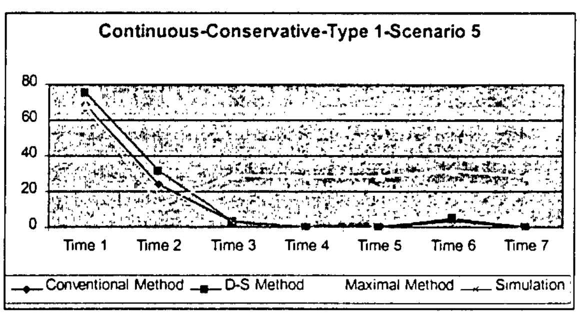
Graphical Comparison of Confidence Values of Proposed Methods and Simulation for Conservative Interpretation

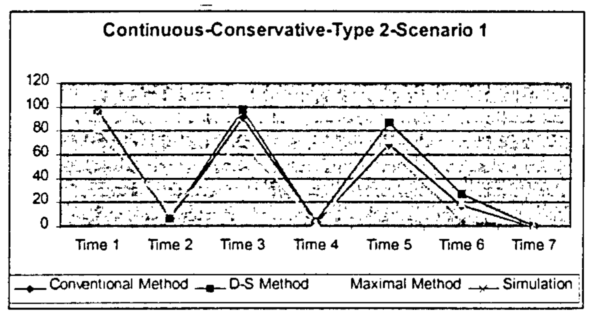


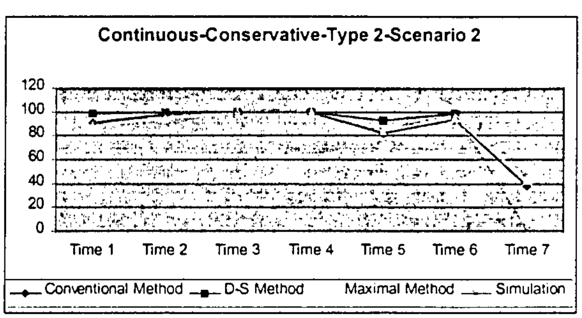


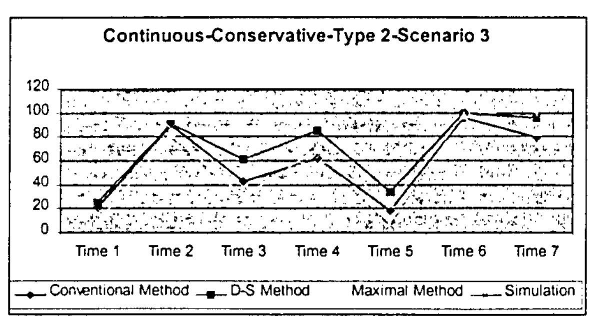


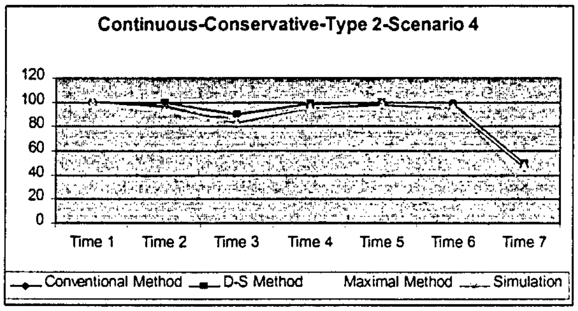


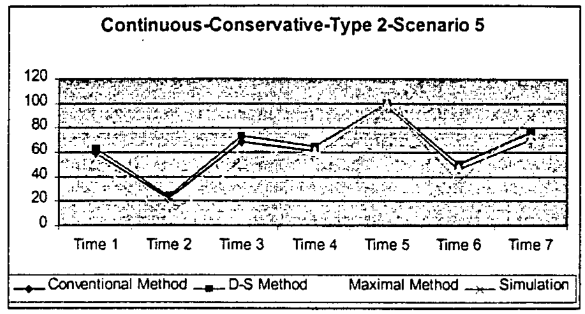


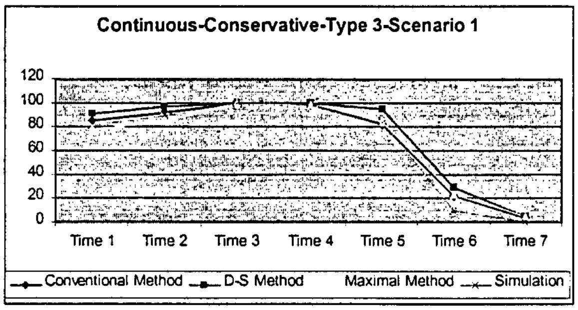


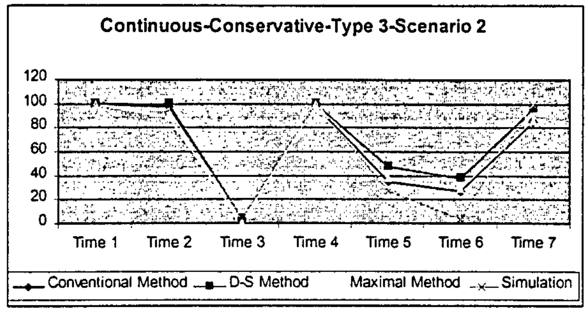


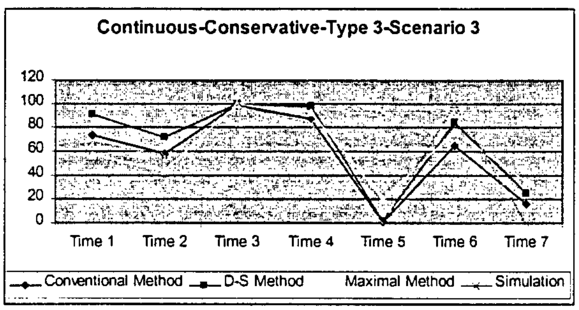


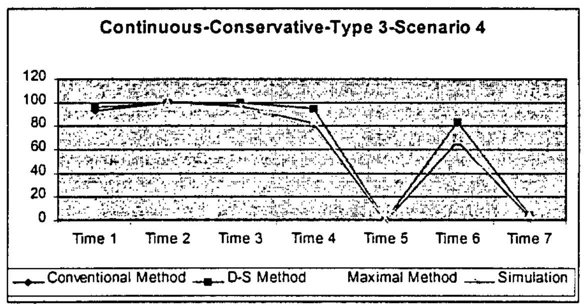


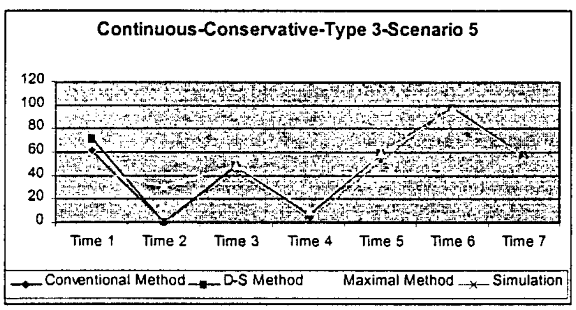












6. Conclusion

The paper presents the methods that assess the performance of an air campaign plan based on ISR information to determine whether the campaign should be continued, re-planned, or terminated. The methods help improving the responsiveness to dynamic battlefield situations by giving recommendations to the commander of the operation. Based on these recommendations, the commander can determine what need to be done to accomplish the air campaign.

Acknowledgement

Dr. Ye would like to acknowledge the funding support to this work that she received from the Air Force Office of Scientific Research through the Summer Research Extension Program (Award No.: 98-0826). The U.S. government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of AFOSR or the U.S. Government.

References

New World Vistas, *Air and Space Power for the 21st Century*, Summary Volume

Jon L. Boyes, Stephen J. Andriole (1987), *Principles of Command and Control.* International Press, Washington DC., AFCEA.

Stephen J. Andriole (1990), *Advanced Technology for Command and Control System Engineering*, AFCEA International Press, Fairfax, VA

Ye, Nong (1997), *A Process Engineering Approach to Continuous Command and Control on Security-Aware Computer Networks*, Final Report for the AFOSR Summer Faculty Research Program/ Rome Laboratory.

Jensen, F. V. (1996), *An Introduction to Bayesian Networks*. UCL Press.

Bender, E. A. (1996), *Mathematical Methods in Artificial Intelligence*. IEEE Computer Society Press

Giarranto , J. , Riley, G. (1994), *Expert Systems: Principles and Programming*, PWS Publishing

Edward L. Waltz, Dennis M. Buede, *Data Fusion and Decision Support for Command and Control*, IEEE Trans. Syst., Man, Cybern., Vol. SMC-16, No. 6, November, 1986

Gordon, J., Shortlife, E. H., *A Method of Managing Evidential Reasoning in a Hierarchical Hypothesis Space*, Artificial Intelligence, Vol. 26, 323-357, 1985.

Filippidis, A., *Data Fusion Using Sensor Data and a Priori Information*, Control Engineering Practice, Vol. 4, No. 1, 43-53, 1996.

Wayne L. Winston (1994), *Operations Research: Applications and Algorithm*, Wadsworth, Inc.

Appendix 1. Scenarios for predicting one task ahead of time

		SCENARIO 1		SCENARIO 2		SCENARIO 3		SCENARIO 4		SCENARIO 5	
		==========		==========		==========		==========		==========	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	4	77	-2	93	1	71	2	79	1	78
	B										
	C										
Time 1	D										
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	-4	79	5	71	1	99	-2	99	-2	87
	C	-5	88	-2	99	-4	75	-3	73	5	86
Time 2	D										
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	93	2	100	-1	70	-3	78	-5	91
	C	1	94	-4	73	-3	97	-5	85	-2	71
Time 3	D	3	76	-2	76	-4	80	3	95	-1	96
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	100	2	100	-1	100	-3	100	-5	100
	C	1	100	-4	100	-3	100	-5	100	-2	100
Time 4	D	5	93	1	74	-4	94	-2	85	-4	73
	E	1	83	4	89	-5	81	0	99	-2	71
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	100	2	100	-1	100	-3	100	-5	100
	C	1	100	-4	100	-3	100	-5	100	-2	100

Time 5	D	5	100	1	100	-4	100	-2	100	-4	100
	E	-2	74	-5	71	-1	98	-4	84	-4	86
	F	5	89	2	75	4	71	-2	89	-4	88
	G										
	H										

		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	100	2	100	-1	100	-3	100	-5	100
	C	1	100	-4	100	-3	100	-5	100	-2	100
Time 6	D	5	100	1	100	-4	100	-2	100	-4	100
	E	-2	100	-5	100	-1	100	-4	100	-4	100
	F	-1	89	5	79	5	100	-1	95	3	74
	G	-2	84	-5	81	2	96	5	97	5	70
	H										

		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-5	100	0	100	1	100	-4	100	-5	100
	B	3	100	2	100	-1	100	-3	100	-5	100
	C	1	100	-4	100	-3	100	-5	100	-2	100
Time 7	D	5	100	1	100	-4	100	-2	100	-4	100
	E	-2	100	-5	100	-1	100	-4	100	-4	100
	F	-1	100	5	100	5	100	-1	100	3	100
	G	-1	100	-1	100	3	100	5	100	0	100
	H	1	77	3	74	-1	94	4	84	-1	85

Appendix 2. Scenarios for predicting two tasks ahead of time

		SCENARIO 1		SCENARIO 2		SCENARIO 3		SCENARIO 4		SCENARIO 5	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	95	5	88	-5	76	4	95	-5	73
	B	1	98	-2	70	-1	90	5	97	4	75
	C	-5	90	1	87	1	94	-2	73	3	81
Time 1	D										
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-4	97	-5	90	3	77	-2	100	-3	76
	C	-4	93	2	95	-3	75	5	87	5	95
Time 2	D	-5	80	5	71	1	81	2	79	-2	78
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-2	99	3	83	-3	93	-2	100	2	83
	C	2	94	5	100	2	76	-3	81	1	93
Time 3	D	1	100	5	84	-1	99	3	74	-2	90
	E	1	83	2	86	-2	81	-2	78	2	80
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-2	100	3	100	-3	100	-2	100	2	100
	C	2	100	5	100	2	100	-3	100	1	100
Time 4	D	1	100	-5	82	-2	93	3	85	5	80
	E	-3	71	5	98	5	100	-2	78	-5	86
	F	-3	71	1	75	-2	96	1	78	-1	90
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-2	100	3	100	-3	100	-2	100	2	100
	C	2	100	5	100	2	100	-3	100	1	100

Time 5	D	1	100	-5	100	-2	100	3	100	5	100
	E	1	79	-3	74	5	100	1	88	2	74
	F	1	95	-3	75	-1	71	-4	73	1	74
	G	-5	81	4	82	-5	97	2	93	2	96
	H										

		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-2	100	3	100	-3	100	-2	100	2	100
	C	2	100	5	100	2	100	-3	100	1	100
Time 6	D	1	100	-5	100	-2	100	3	100	5	100
	E	1	100	-3	100	5	100	0	100	2	100
	F	-5	100	3	100	0	100	-4	100	-4	100
	G	1	89	3	73	3	85	2	79	-1	94
	H	-4	99	-1	95	3	73	2	90	-3	86

		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	0	100	4	100	-1	100	3	100	-5	100
	B	-2	100	3	100	-3	100	-2	100	2	100
	C	2	100	5	100	2	100	-3	100	1	100
Time 7	D	1	100	-5	100	-2	100	3	100	5	100
	E	1	100	-3	100	5	100	0	100	2	100
	F	-5	100	3	100	0	100	-4	100	-4	100
	G	-4	100	0	100	3	100	-4	100	-1	100
	H	-3	75	-4	78	1	70	2	86	2	84

Appendix 3. Scenarios for predicting three tasks ahead of time

		SCENARIO 1		SCENARIO 2		SCENARIO 3		SCENARIO 4		SCENARIO 5	
		===========		===========		===========		===========		===========	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	-1	74	4	77	-3	87	-4	77	4	73
	B	-1	77	5	98	2	87	3	74	-5	94
	C	-1	76	1	79	5	96	1	76	4	85
Time 1	D	2	95	5	74	-1	88	3	94	-5	70
	E										
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	85	1	98	-5	99	5	84	-5	95
	C	-2	98	4	75	4	81	-1	89	-3	75
Time 2	D	1	88	2	90	4	71	1	90	-3	77
	E	1	82	3	82	-4	84	1	83	-5	81
	F										
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	100	-2	100	-5	100	-3	100	2	100
	C	1	71	5	99	-1	81	1	95	-3	75
Time 3	D	5	76	-5	98	5	95	-1	87	1	95
	E	2	82	-4	89	3	94	4	99	-4	99
	F	2	70	-2	89	2	73	-5	83	-4	84
	G										
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	100	-2	100	-5	100	-3	100	2	100
	C	1	100	5	100	-1	100	0	100	-3	100
Time 4	D	1	80	1	92	-4	72	-5	84	-3	79
	E	-2	75	-1	88	2	99	-1	89	1	80
	F	4	70	4	84	2	73	-2	76	-5	78
	G	2	89	5	92	1	98	5	81	-4	90
	H										
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	100	-2	100	-5	100	-3	100	2	100
	C	1	100	5	100	-1	100	0	100	-3	100

Time 5	D	1	100	0	100	-4	100	-5	100	-3	100
	E	-3	100	3	100	2	100	-4	100	1	100
	F	-5	70	-4	80	5	78	-5	89	-5	84
	G	5	89	2	96	-5	79	1	98	5	88
	H	1	95	-3	72	-4	87	-5	81	-2	79
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	100	-2	100	-5	100	-3	100	2	100
	C	1	100	5	100	-1	100	0	100	-3	100
Time 6	D	1	100	0	100	-4	100	-5	100	-3	100
	E	-3	100	3	100	2	100	-4	100	1	100
	F	-5	100	5	100	4	100	3	100	3	100
	G	1	83	-5	93	-2	86	5	72	1	78
	H	1	76	-3	73	4	95	-1	91	4	73
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
	A	1	100	-5	100	-2	100	4	100	-2	100
	B	-2	100	-2	100	-5	100	-3	100	2	100
	C	1	100	5	100	-1	100	0	100	-3	100
Time 7	D	1	100	0	100	-4	100	-5	100	-3	100
	E	-3	100	3	100	2	100	-4	100	1	100
	F	-5	100	5	100	4	100	3	100	3	100
	G	-4	100	5	100	-2	100	-4	100	2	100
	H	3	93	-3	100	1	91	1	72	-3	79

Appendix 4. Results of Discrete Interpretation: Type 1

		Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
Time 1	Method 1	3.08	77.00	0.00	0.00	0.71	71.00	1.58	79.00	0.78	78.00
	Method 2	3.08	77.00	0	0.00	0.71	71.00	1.58	79.00	0.78	78.00
	Method 3										
	Simulation	4.00	77.00	0.00	0.00	1.00	71.00	2.00	78.10	1.00	79.20
Time 2	Method 1	0.00	0.00	3.55	71.00	1.99	100.00	0.00	0.00	0.00	0.00
	Method 2	0	0.00	3.55	71.00	1.99	100.00	0	0.00	0	0.00
	Method 3										
	Simulation	0.00	0.00	5.00	69.30	1.99	100.00	2.00	0.90	0	0
Time 3	Method 1	0.71	70.68	0.48	24.00	0.06	6.00	0.00	0.00	0.00	0.00
	Method 2	0.71	70.68	0.48	24.00	0.06	6.00	0	0.00	0	0.00
	Method 3										
	Simulation	1.00	71.00	2.00	23.40	1.00	6.10	0.00	0.00	0.00	0.00
Time 4	Method 1	3.56	93.00	6.30	100.00	0.00	0.00	0.00	0.00	0.00	0.00
	Method 2	3.56	93.00	6.3	100.00	0	0.00	0	0.00	0	0.00
	Method 3										
	Simulation	3.82	93.00	6.36	100.00	0.00	0.00	0.00	0.00	0.00	0.00
Time 5	Method 1	5.97	100.00	1.31	29.00	0.00	0.00	0.00	0.00	0.00	0.00
	Method 2	5.97	100.00	1.31	29.00	0	0.00	0	0.00	0	0.00
	Method 3										
	Simulation	6.05	100.00	4.49	27.50	0.00	0.00	0.00	0.00	0.00	0.00
Time 6	Method 1	0.02	1.76	0.45	15.01	1.92	96.00	0.00	0.00	0.00	0.00
	Method 2	0.02	1.76	0.45	15.01	1.92	96.00	0	0.00	0	0.00
	Method 3										
	Simulation	1.00	1.10	3.00	15.80	2.00	96.30	0.00	0.00	0.00	0.00
Time 7	Method 1	0.00	0.00	4.22	100.00	2.06	100.00	0.00	0.00	0.00	0.00
	Method 2	0	0.00	4.22	100.00	2.06	100.00	0	0.00	0	0.00
	Method 3										
	Simulation	0.00	0.00	4.26	100.00	2.07	100.00	0.00	0.00	0.00	0.00

Appendix 5. Results of Discrete Interpretation: Type 2

		Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
Time 1	Method 1	1.93	99.90	3.17	88.00	0.00	0.00	8.65	99.85	0.81	20.25
	Method 2	1.93	99.90	3.17	88.00	0.00	0.00	8.65	99.85	0.81	20.25
	Method 3										
	Simulation	1.93	99.90	3.65	89.10	0.00	0.00	8.65	99.90	4.00	19.40
Time 2	Method 1	0.00	0.00	4.46	98.55	2.16	77.00	3.98	100.00	0.00	0.00
	Method 2	0	0.00	4.46	99.62	2.16	77.00	3.98	100.00	0	0.00
	Method 3										
	Simulation	0.00	0.00	4.62	98.30	2.81	76.10	4.56	100.00	0.00	0.00
Time 3	Method 1	0.02	1.00	12.41	100.00	0.00	0.00	1.86	79.72	0.00	0.00
	Method 2	0.02	1.00	12.41	100.00	0	0.00	1.86	80.35	0	0.00
	Method 3										
	Simulation	1.73	1.10	12.45	100.00	0.00	0.00	2.36	79.50	0.00	0.00
Time 4	Method 1	0.00	0.00	8.55	100.00	0.00	0.28	2.80	88.30	0.12	11.20
	Method 2	0	0.00	8.55	100.00	0	0.56	2.8	88.30	0.12	11.20
	Method 3										
	Simulation	0.00	0.00	8.57	100.00	1.00	0.50	3.17	88.40	1.06	10.90
Time 5	Method 1	0.14	14.26	1.28	37.66	0.00	0.00	3.82	99.39	6.14	100.00
	Method 2	0.14	14.26	1.28	40.98	0	0.00	3.82	99.52	6.14	100.00
	Method 3										
	Simulation	1.00	14.80	3.40	37.40	0.00	0.00	3.93	99.40	6.10	100.00
Time 6	Method 1	0.00	0.00	3.24	100.00	3.78	95.95	3.38	97.90	0.00	0.00
	Method 2	0	0.00	3.24	100.00	3.78	99.84	3.38	97.90	0	0.00
	Method 3										
	Simulation	0.00	0.00	3.17	100.00	3.93	96.60	3.47	97.90	0.00	0.00
Time 7	Method 1	0.00	0.00	0.44	22.00	2.70	100.00	0.00	0.00	0.84	84.00
	Method 2	0	0.00	0.44	22.00	2.7	100.00	0	0.00	0.84	84.00
	Method 3										
	Simulation	0.00	0.00	2.00	20.80	2.67	100.00	0.00	0.00	1.00	85.00

Appendix 6. Results of Discrete Interpretation: Type 3

		Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
		MoM	%	MoM	%	MoM	%	MoM	%	MoM	%
Time 1	Method 1	0.47	40.87	11.68	99.88	0.13	11.31	2.23	76.20	0.56	18.62
	Method 2	0.47	40.87	11.68	99.96	0.13	11.31	2.23	76.20	0.56	19.68
	Method 3										
	Simulation	1.16	40.60	11.20	99.30	1.15	12.00	2.97	76.30	4.00	1.30
Time 2	Method 1	1.02	76.34	0.72	72.32	0.09	9.20	9.93	100.00	0.00	0.00
	Method 2	1.02	76.34	0.72	72.32	0.09	9.30	9.93	100.00	0	0.00
	Method 3										
	Simulation	1.31	75.40	1.00	73.50	2.00	10.40	9.92	100.00	3.62	5.20
Time 3	Method 1	5.85	98.70	0.00	0.00	2.20	89.30	0.70	16.85	0.00	0.15
	Method 2	5.85	99.82	0	0.00	2.2	96.77	0.7	30.85	0	0.15
	Method 3										
	Simulation	5.89	98.80	0.00	0.00	2.45	89.70	4.18	16.20	1.00	0.30
Time 4	Method 1	2.98	76.68	3.14	78.91	0.00	0.00	0.48	14.84	0.00	0.37
	Method 2	2.98	91.66	3.14	95.21	0	0.00	0.48	25.87	0	0.37
	Method 3										
	Simulation	3.81	77.10	3.97	80.50	0.00	0.00	3.09	17.00	1.00	0.20
Time 5	Method 1	0.79	26.70	0.00	0.00	0.00	0.00	0.00	0.00	0.20	14.08
	Method 2	0.79	45.29	0	0.00	0	0.00	0	0.00	0.2	14.08
	Method 3										
	Simulation	2.97	26.00	0.00	0.00	0.00	0.00	0.00	0.00	1.36	15.60
Time 6	Method 1	0.00	0.00	0.06	1.89	0.00	0.00	0.00	0.00	4.70	100.00
	Method 2	0	0.00	0.06	3.74	0	0.00	0	0.00	4.7	100.00
	Method 3										
	Simulation	0.00	0.00	3.00	2.20	0.00	0.00	0.00	0.00	4.68	100.00
Time 7	Method 1	0.00	0.00	5.00	100.00	0.00	0.00	0.00	0.00	0.63	21.00
	Method 2	0	0.00	5	100.00	0	0.00	0	0.00	0.63	21.00
	Method 3										
	Simulation	0.00	0.00	5.00	100.00	0.00	0.00	0.00	0.00	3.00	21.80

Appendix 7. Result of Optimist Interpretation: Type 1

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	3.08	2	67.28	-1.86	-1	40.55	0.71	0.50	54.77	1.58	1.00	59.36	0.78	0.50	54.78
	Method 2	3.08	2	75.13	-1.86	-1	44.36	0.71	0.50	60.48	1.58	1.00	65.77	0.78	0.50	60.46
	Method 3	3.08	2	67.41	-1.86	-1	40.65	0.71	0.50	54.89	1.58	1.00	59.49	0.78	0.50	54.90
	Simulation	4.34		67.40	2.85		39.60	3.52		57.20	3.80		59.80	3.35		55.90
Time 2	Method 1	-8.16	-7	4.29	3.55	2.50	70.81	1.99	1.50	64.79	-5.98	-5	9.37	-0.70	-6	5.75
	Method 2	-8.16	-7	4.40	3.55	2.50	72.65	1.99	1.50	67.68	-5.98	-5	9.77	-0.70	-6	8.98
	Method 3	-8.16	-7	5.31	3.55	2.50	71.24	1.99	1.50	64.79	-5.98	-5	9.37	-0.70	-6	6.52
	Simulation	1.68		4.50	4.56		73.10	3.71		66.10	2.05		9.00	1.49		5.70
Time 3	Method 1	0.07	-2	30.17	0.48	1.00	59.09	-2.90	-1.50	35.67	-3.49	-4	14.88	-7.38	-8	1.86
	Method 2	0.07	-2	31.35	0.48	1.00	60.26	-2.90	-1.50	37.86	-3.49	-4	15.45	-7.38	-8	1.94
	Method 3	0.07	-2.00	30.17	0.48	1.00	59.43	-2.90	-1.50	35.67	-3.49	-4	14.92	-7.38	-8	2.97
	Simulation	2.32		27.60	3.31		58.50	2.48		36.40	2.00		16.10	1.10		1.60
Time 4	Method 1	3.48	1	59.85	6.30	4.50	86.65	-7.81	-4.50	14.65	-8.70	-8	1.02	-11.34	-13	0.04
	Method 2	3.48	1	61.75	6.30	4.50	87.09	-7.81	-4.50	14.83	-8.70	-8	1.02	-11.34	-13	0.04
	Method 3	3.48	1	60.83	6.30	4.50	86.69	-7.81	-4.50	15.06	-8.70	-8	5.02	-11.34	-13	28.27
	Simulation	2.99		62.00	5.01		93.40	1.70		10.30	0.97		0.20	0.00		0.00
Time 5	Method 1	5.97	4.50	86.05	0.95	1.50	63.33	-2.14	-2.50	25.36	-14.14	-12	0.02	-17.96	-18	0.00
	Method 2	5.97	4.50	89.58	0.95	1.50	63.62	-2.14	-2.50	25.58	-14.14	-12	0.02	-17.96	-18	0.00
	Method 3	5.97	4.50	86.21	0.95	1.50	63.61	-2.14	-2.50	25.77	-14.14	-12	4.02	-17.96	-18	28.24
	Simulation	5.04		94.50	3.31		70.10	1.72		19.60	0.00		0.00	0.00		0.00
Time 6	Method 1	-1.57	-0.50	44.64	-2.10	-2	32.19	1.92	1.00	59.69	-9.10	-11	0.01	-9.28	-14	0.01
	Method 2	-1.57	-0.50	45.23	-2.10	-2	32.21	1.92	1.00	61.96	-9.10	-11	0.01	-9.28	-14	0.01
	Method 3	-1.57	-0.50	46.35	-2.10	-2	33.27	1.92	1.00	60.04	-9.10	-11	4.01	-9.28	-14	28.25
	Simulation	1.00		36.60	1.65		28.00	1.52		80.80	0.00		0.00	0.00		0.00
Time 7	Method 1	-0.23	-0.50	45.30	4.22	3.50	77.96	2.06	2.50	71.21	-5.64	-7	4.50	-12.85	-15.5	0.00
	Method 2	-0.23	-0.50	46.68	4.22	3.50	78.69	2.06	2.50	75.30	-5.64	-7	4.51	-12.85	-15.5	0.00
	Method 3	-0.23	-0.50	47.52	4.22	3.50	78.08	2.06	2.50	71.51	-5.64	-7	8.48	-12.85	-15.5	28.24
	Simulation	0.23		10.80	3.49		99.80	2.51		100.00	0.00		0.00	0.00		0.00

Appendix 8. Result of Optimist Interpretation: Type 2

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	1.93	1.00	60.62	5.27	1.50	64.13	-2.86	-3	24.15	8.65	4.50	86.94	-0.65	-0.50	45.74
	Method 2	1.93	1.00	62.20	5.27	1.50	75.27	-2.86	-3	28.00	8.65	4.50	88.97	-0.65	-0.50	50.65
	Method 3	1.93	1.00	60.62	5.27	1.50	64.13	-2.86	-3	24.15	8.65	4.50	86.95	-0.65	-0.50	46.08
	Simulation	3.47		62.20	4.00		64.50	2.30		23.50	5.41		87.20	3.70		45.60
Time 2	Method 1	-7.72	-4.5	14.66	9.45	4.00	79.75	2.12	1.00	60.16	8.93	2.00	68.12	-1.81	-7.5	2.56
	Method 2	-7.72	-4.5	15.85	9.45	4.00	94.07	2.12	1.00	61.61	8.93	2.00	88.45	-1.81	-7.5	3.88
	Method 3	-7.72	-4.50	14.66	9.45	4.00	80.93	2.12	1.00	60.16	8.93	2.00	74.21	-1.81	-7.50	2.93
	Simulation	1.80		14.20	5.46		82.40	3.48		61.60	3.78		73.90	1.23		2.50
Time 3	Method 1	3.71	0.50	55.66	14.92	9.00	98.67	-2.09	-4	11.56	3.66	3.50	74.35	-3.54	-4	10.85
	Method 2	3.71	0.50	66.28	14.92	9.00	99.94	-2.09	-4	13.87	3.66	3.50	79.97	-3.54	-4	10.97
	Method 3	3.71	0.50	55.66	14.92	9.00	98.92	-2.09	-4	11.59	3.66	3.50	74.35	-3.54	-4	10.85
	Simulation	2.44		54.90	9.01		99.20	1.34		10.20	3.37		70.30	1.35		11.70
Time 4	Method 1	-1.26	-4	15.24	10.55	7.50	96.14	2.22	-1	39.94	4.77	4.00	86.50	-4.20	-3.5	19.62
	Method 2	-1.26	-4	23.45	10.55	7.50	99.48	2.22	-1	63.93	4.77	4.00	87.95	-4.20	-3.5	20.09
	Method 3	-1.26	-4	17.64	10.55	7.50	97.05	2.22	-1	48.51	4.77	4.00	86.60	-4.20	-3.50	21.25
	Simulation	1.57		9.70	7.55		98.90	1.47		33.80	3.01		75.20	1.74		15.60
Time 5	Method 1	0.69	-2.5	23.07	2.81	1.00	59.94	-1.56	-4	14.15	5.82	5.50	92.88	6.14	4.50	91.06
	Method 2	0.69	-2.5	34.38	2.81	1.00	71.88	-1.56	-4	26.29	5.82	5.50	94.51	6.14	4.50	93.78
	Method 3	0.69	-2.50	26.75	2.81	1.00	63.32	-1.56	-4	19.38	5.82	5.50	92.96	6.14	4.50	91.16
	Simulation	1.12		13.50	2.51		64.40	0.49		0.90	3.91		95.10	4.53		99.70
Time 6	Method 1	-4.07	-6.5	3.22	5.24	3.00	76.18	8.74	2.00	68.51	5.38	4.00	84.75	-3.52	-2	30.82
	Method 2	-4.07	-6.5	4.68	5.24	3.00	88.09	8.74	2.00	90.08	5.38	4.00	86.71	-3.52	-2	31.80
	Method 3	-4.07	-6.5	3.88	5.24	3.00	78.85	8.74	2.00	75.91	5.38	4.00	84.85	-3.52	-2	31.81
	Simulation	0.00		0.00	3.09		99.10	2.36		86.90	2.06		98.30	0.40		2.30
Time 7	Method 1	-7.25	-10.5	0.36	0.88	0.00	50.00	7.70	2.50	71.40	-0.28	-1	40.29	0.68	0.00	50.00
	Method 2	-7.25	-10.5	0.50	0.88	0.00	62.68	7.70	2.50	91.82	-0.28	-1	40.75	0.68	0.00	53.27
	Method 3	-7.25	-10.50	0.48	0.88	0.00	54.09	7.70	2.50	78.03	-0.28	-1	40.43	0.68	0.00	50.53
	Simulation	0.00		0.00	1.31		47.80	2.48		100.00	0.85		32.7	0.59		48.80

Appendix 9. Result of Optimist Interpretation: Type 3

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	0.40	0.00	50.00	11.28	7.00	93.17	1.31	-1	39.32	1.96	1.00	59.50	2.82	-3	26.57
	Method 2	0.40	0.00	54.03	11.28	7.00	96.58	1.31	-1	46.61	1.96	1.00	63.69	2.82	-3	43.71
	Method 3	0.40	0.00	50.00	11.28	7.00	93.18	1.31	-1	39.35	1.96	1.00	59.51	2.82	-3	30.23
	Simulation	2.85		51.00	7.50		92.70	2.56		39.60	3.68		59.30	3.09		27.30
Time 2	Method 1	1.00	1.00	61.74	2.26	-2	27.00	0.72	-4.5	13.13	9.93	7.50	97.34	-10.61	-8.5	1.80
	Method 2	1.00	1.00	66.06	2.26	-2	31.97	0.72	-4.5	23.22	9.93	7.50	98.52	-10.61	-8.5	1.90
	Method 3	1.00	1.00	61.74	2.26	-2	27.16	0.72	-4.5	15.99	9.93	7.50	97.34	-10.61	-8.50	2.33
	Simulation	2.95		64.50	2.00		26.00	2.01		14.70	7.53		98.20	3.93		1.10
Time 3	Method 1	8.55	3.50	81.21	-10.29	-12.5	0.01	6.22	-2	27.07	3.89	0.00	50.00	-6.37	-3.5	16.85
	Method 2	8.55	3.50	92.64	-10.29	-12.5	0.01	6.22	-2	42.11	3.89	0.00	66.70	-6.37	-3.5	16.86
	Method 3	8.55	3.50	81.77	-10.29	-12.50	0.01	6.22	-2	38.96	3.89	0.00	51.23	-6.37	-3.50	18.40
	Simulation	4.26		85.60	0.00		0.00	1.77		23.70	2.34		48.50	1.37		9.60
Time 4	Method 1	5.88	1.50	67.36	8.00	-0.5	43.63	-1.46	-6.5	1.32	1.44	-0.5	44.78	-9.07	-5.5	7.29
	Method 2	5.88	1.50	81.67	8.00	-0.5	54.92	-1.46	-6.5	1.77	1.44	-0.5	58.62	-9.07	-5.5	7.29
	Method 3	5.88	1.50	69.98	8.00	-0.50	46.99	-1.46	-6.50	20.89	1.44	-0.50	47.30	-9.07	-5.50	9.89
	Simulation	2.78		71.10	1.78		42.10	1.59		0.40	2.07		42.00	1.33		3.80
Time 5	Method 1	1.90	-2.5	27.22	-0.44	-4.5	12.37	-8.53	-11	0.43	-12.52	-12.5	0.04	-3.38	-3	23.12
	Method 2	1.90	-2.5	36.83	-0.44	-4.5	16.10	-8.53	-11	0.62	-12.52	-12.5	0.04	-3.38	-3	23.13
	Method 3	1.90	-2.50	28.62	-0.44	-4.50	13.34	-8.53	-11	19.90	-12.52	-12.50	1.27	-3.38	-3	24.54
	Simulation	1.87		19.20	1.13		1.50	0.91		0.10	0.00		0.00	1.41		11.00
Time 6	Method 1	-3.41	-7	1.56	1.16	-1	41.27	1.08	-4	14.91	0.69	-3	24.03	4.70	3.50	79.02
	Method 2	-3.41	-7	1.76	1.16	-1	55.91	1.08	-4	22.95	0.69	-3	32.80	4.70	3.50	80.01
	Method 3	-3.41	-7	1.77	1.16	-1	43.62	1.08	-4	34.23	0.69	-3	26.05	4.70	3.50	79.12
	Simulation	0.00		0.00	1.20		30.70	0.00		0.00	1.19		11.00	3.70		97.00
Time 7	Method 1	-6.21	-10.5	0.17	10.00	5.00	84.73	-2.09	-6.5	5.21	-5.28	-8.5	1.35	0.63	1.50	62.74
	Method 2	-6.21	-10.5	0.18	10.00	5.00	95.95	-2.09	-6.5	7.98	-5.28	-8.5	1.60	0.63	1.50	63.53
	Method 3	-6.21	-10.5	0.33	10.00	5.00	86.57	-2.09	-6.5	24.78	-5.28	-8.5	2.63	0.63	1.50	63.07
	Simulation	0.00		0.00	5.00		100.00	0.00		0.00	0.00		0.00	1.72		89.70

Appendix 10. Results of Conservative Interpretation: Type 1

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	3.08	5.29	88.02	-1.86	-0.524	45.11	0.71	1.76	65.66	1.58	3.21	76.67	0.78	2.06	68.15
	Method 2	3.08	5.29	94.34	-1.86	-0.524	49.78	0.71	1.76	73.11	1.58	3.21	84.47	0.78	2.06	75.76
	Method 3	3.08	4.00	88.14	-1.86	-2	45.22	0.71	1.00	65.79	1.58	2.00	76.80	0.78	1.00	68.29
	Simulation	6.45		84.90	3.55		41.40	4.47		61.50	5.20		73.60	4.56		65.00
Time 2	Method 1	-8.16	-7.891	2.41	3.55	6.38	91.58	1.99	6.65	93.37	-5.98	-2.22	29.45	-0.70	-2.974	23.90
	Method 2	-8.16	-7.891	2.42	3.55	6.38	92.61	1.99	6.65	93.58	-5.98	-2.22	29.56	-0.70	-2.974	31.74
	Method 3	-8.16	-9.00	13.05	3.55	5.00	87.34	1.99	2.00	67.52	-5.98	-6.00	8.48	-0.70	-5.00	12.56
	Simulation	2.25		1.60	7.37		88.30	7.49		90.10	3.09		26.60	2.88		21.30
Time 3	Method 1	0.07	5.47	88.81	0.48	0.71	56.40	-2.90	-2.451	27.97	-3.49	-0.169	48.39	-7.38	-7.355	3.02
	Method 2	0.07	5.47	91.67	0.48	0.71	56.66	-2.90	-2.451	28.90	-3.49	-0.169	49.75	-7.38	-7.355	3.33
	Method 3	0.07	1.00	69.19	0.48	0.00	56.80	-2.90	-4.00	21.52	-3.49	-4.00	41.08	-7.38	-11.00	29.23
	Simulation	6.90		85.00	3.66		54.20	2.99		25.60	3.80		46.90	2.00		2.70
Time 4	Method 1	3.48	8.68	96.91	6.30	10.80	98.90	-7.81	-6.737	4.44	-8.70	-3.602	18.34	-11.34	-14.64	0.00
	Method 2	3.48	8.68	98.93	6.30	10.80	99.43	-7.81	-6.737	4.45	-8.70	-3.602	18.53	-11.34	-14.64	0.00
	Method 3	3.48	4.00	96.95	6.30	7.00	98.91	-7.81	-9.00	4.85	-8.70	-9.00	22.20	-11.34	-16.00	28.24
	Simulation	8.92		97.70	10.75		98.80	1.71		1.70	2.45		15.70	0.00		0.00
Time 5	Method 1	5.97	10.11	98.26	0.95	2.13	68.65	-2.14	3.91	81.32	-14.14	-11.64	0.07	-17.96	-18.9	0.00
	Method 2	5.97	10.11	99.58	0.95	2.13	69.00	-2.14	3.91	85.55	-14.14	-11.64	0.07	-17.96	-18.9	0.00
	Method 3	5.97	6.00	98.29	0.95	0.00	68.85	-2.14	-1.00	81.57	-14.14	-15.00	4.08	-17.96	-22.00	28.24
	Simulation	10.27		98.80	4.04		68.40	4.40		65.10	1.00		0.10	0.00		0.00
Time 6	Method 1	-1.57	1.76	65.75	-2.10	1.11	60.11	1.92	5.72	89.45	-9.10	-1.42	36.37	-9.28	-7.241	4.08
	Method 2	-1.57	1.76	68.94	-2.10	1.11	60.32	1.92	5.72	93.50	-9.10	-1.42	37.26	-9.28	-7.241	5.22
	Method 3	-1.57	-2.00	66.64	-2.10	-2.00	60.42	1.92	2.00	89.64	-9.10	-9.00	39.92	-9.28	-10.00	31.93
	Simulation	3.47		69.00	3.41		59.90	5.85		96.00	3.13		31.50	2.36		2.80
Time 7	Method 1	-0.23	1.02	59.07	4.22	6.05	90.39	2.06	3.75	79.25	-5.64	-3.26	22.16	-12.85	-14.83	0.00
	Method 2	-0.23	1.02	61.94	4.22	6.05	91.53	2.06	3.75	83.96	-5.64	-3.26	22.59	-12.85	-14.83	0.00
	Method 3	-0.23	0.00	60.39	4.22	5.00	90.43	2.06	2.00	79.51	-5.64	-5.00	25.96	-12.85	-16.00	28.24
	Simulation	1.88		61.50	6.07		99.90	3.83		99.00	1.65		3.10	0.00		0.00

Appendix 11. Results of Conservative Interpretation: Type 2

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	1.93	8.37	97.08	5.27	6.06	91.31	-2.86	-3.316	21.15	8.65	16.58	99.97	-0.65	0.98	58.78
	Method 2	1.93	8.37	97.28	5.27	6.06	98.80	-2.86	-3.316	25.40	8.65	16.58	99.99	-0.65	0.98	62.99
	Method 3	1.93	2.00	83.28	5.27	3.00	89.15	-2.86	-6.00	9.15	8.65	9.00	99.35	-0.65	-1.00	50.96
	Simulation	8.77		95.60	7.04		89.20	3.02		19.50	16.46		99.70	4.27		55.00
Time 2	Method 1	-7.72	-5.888	7.02	9.45	9.50	98.11	2.12	5.87	90.35	8.93	8.32	96.69	-1.81	-3.144	22.30
	Method 2	-7.72	-5.888	7.08	9.45	9.50	99.87	2.12	5.87	90.94	8.93	8.32	99.43	-1.81	-3.144	23.84
	Method 3	-7.72	-9.00	13.53	9.45	6.00	94.63	2.12	3.00	82.74	8.93	5.00	92.92	-1.81	-7.00	6.76
	Simulation	2.26		5.70	9.73		97.30	7.00		87.40	8.84		95.50	2.85		20.40
Time 3	Method 1	3.71	5.69	90.59	14.92	20.31	100.00	-2.09	-0.743	42.64	1.66	4.30	83.71	-3.54	2.10	68.48
	Method 2	3.71	5.69	97.89	14.92	20.31	100.00	-2.09	-0.743	60.87	1.66	4.30	90.41	-3.54	2.10	73.76
	Method 3	3.71	0.00	71.34	14.92	14.00	99.99	-2.09	-7.00	29.61	1.66	2.00	84.33	-3.54	-3.00	51.17
	Simulation	6.87		87.50	20.45		100.00	3.33		38.90	5.51		85.10	4.85		63.10
Time 4	Method 1	-1.26	-7.643	2.69	10.55	15.06	99.90	2.22	1.32	62.06	2.77	7.48	95.43	-4.20	1.17	60.84
	Method 2	-1.26	-7.643	4.01	10.55	15.06	100.00	2.22	1.32	85.61	2.77	7.48	98.35	-4.20	1.17	65.15
	Method 3	-1.26	-7.00	3.23	10.55	8.00	99.94	2.22	-3.00	69.73	2.77	3.00	95.71	-4.20	-4.00	61.16
	Simulation	2.03		3.70	15.24		99.80	3.39		62.50	7.83		96.60	3.87		57.10
Time 5	Method 1	0.69	2.00	68.51	2.81	4.15	82.20	-1.56	-3.681	18.57	3.82	9.33	98.18	6.14	13.37	99.73
	Method 2	0.69	2.00	87.29	2.81	4.15	92.97	-1.56	-3.681	33.69	3.82	9.33	99.61	6.14	13.37	99.99
	Method 3	0.69	-4.00	72.93	2.81	0.00	84.41	-1.56	-7.00	24.23	3.82	3.00	98.32	6.14	7.00	99.74
	Simulation	4.25		63.80	5.35		83.90	1.74		5.80	9.40		99.00	13.61		99.90
Time 6	Method 1	-4.07	-3.802	17.26	5.24	7.23	93.77	8.74	8.33	96.21	3.38	7.64	95.58	-3.52	-0.334	46.92
	Method 2	-4.07	-3.802	26.66	5.24	7.23	98.91	8.74	8.33	99.86	3.38	7.64	98.45	-3.52	-0.334	49.94
	Method 3	-4.07	-8.00	19.83	5.24	4.00	94.87	8.74	5.00	97.69	3.38	4.00	95.84	-3.52	-4.00	47.53
	Simulation	1.72		3.60	7.20		99.10	8.25		99.50	7.78		98.90	2.04		36.00
Time 7	Method 1	-7.25	-11.41	0.13	0.88	-1.421	37.51	7.70	3.72	79.03	-2.28	-0.38	46.38	0.68	2.49	71.06
	Method 2	-7.25	-11.41	0.17	0.88	-1.421	46.95	7.70	3.72	95.60	-2.28	-0.38	49.23	0.68	2.49	76.88
	Method 3	-7.25	-12.00	0.20	0.88	-2.00	42.37	7.70	3.00	84.40	-2.28	-2.00	47.51	0.68	1.00	71.30
	Simulation	0.00		0.00	1.00		0.60	3.81		98.30	1.58		30.50	2.91		88.30

Appendix 12. Results of Conservative Interpretation: Type 3

		Scenario 1			Scenario 2			Scenario 3			Scenario 4			Scenario 5		
		MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%	MoM 1	MoM 2	%
Time 1	Method 1	0.40	4.48	85.02	11.68	21.63	100.00	1.31	2.70	73.41	1.96	6.53	92.59	2.82	1.21	61.08
	Method 2	0.40	4.48	91.05	11.68	21.63	100.00	1.31	2.70	90.63	1.96	6.53	95.63	2.82	1.21	71.21
	Method 3	0.40	0.00	77.21	11.68	14.00	99.95	1.31	-2.00	57.06	1.96	2.00	87.17	2.82	-2.00	44.77
	Simulation	5.97		81.50	19.42		100.00	5.02		67.60	7.47		89.40	4.21		58.40
Time 2	Method 1	1.00	6.13	92.01	2.26	9.09	97.82	0.72	0.87	58.23	9.93	17.16	99.98	-10.61	-11.2	0.11
	Method 2	1.00	6.13	97.13	2.26	9.09	99.62	0.72	0.87	72.05	9.93	17.16	100.00	-10.61	-11.2	0.11
	Method 3	1.00	1.00	85.18	2.26	1.00	88.79	0.72	-3.00	44.01	9.93	11.00	99.87	-10.61	-15.00	30.76
	Simulation	7.08		89.10	9.28		96.00	3.94		56.10	17.13		100.00	1.50		0.20
Time 3	Method 1	8.55	12.38	99.57	-10.29	-6.675	3.67	6.22	10.88	99.16	3.89	8.25	97.24	-6.37	-0.172	48.30
	Method 2	8.55	12.38	100.00	-10.29	-6.675	3.69	6.22	10.88	99.99	3.89	8.25	99.85	-6.37	-0.172	48.47
	Method 3	8.55	8.00	99.67	-10.29	-16.00	0.28	6.22	3.00	99.61	3.89	-1.00	95.95	-6.37	-7.00	48.76
	Simulation	12.45		99.40	1.94		3.60	11.02		98.60	8.85		95.40	3.27		42.30
Time 4	Method 1	5.88	10.11	98.67	8.00	13.86	99.83	-1.46	4.79	87.24	1.44	4.11	82.58	-9.07	-5.884	6.62
	Method 2	5.88	10.11	99.95	8.00	13.86	100.00	-1.46	4.79	97.68	1.44	4.11	94.56	-9.07	-5.884	6.62
	Method 3	5.88	4.00	98.93	8.00	4.00	99.89	-1.46	-6.00	92.50	1.44	-2.00	84.24	-9.07	-11.00	9.39
	Simulation	10.03		98.90	13.93		99.80	6.03		83.40	5.63		78.70	1.57		3.00
Time 5	Method 1	1.90	3.98	82.36	-0.44	-1.613	35.08	-8.53	-9.217	1.06	-12.52	-10.81	0.17	-3.38	0.99	58.93
	Method 2	1.90	3.98	94.39	-0.44	-1.613	47.36	-8.53	-9.217	1.56	-12.52	-10.81	0.18	-3.38	0.99	59.37
	Method 3	1.90	-2.00	83.99	-0.44	-7.00	37.34	-8.53	-13.00	20.57	-12.52	-17.00	1.40	-3.38	-4.00	59.28
	Simulation	5.06		84.40	2.35		27.30	1.00		0.20	0.00		0.00	3.32		52.10
Time 6	Method 1	-3.41	-3.121	21.86	1.16	-2.619	27.45	1.08	1.63	64.62	0.69	1.97	66.92	4.70	8.62	96.56
	Method 2	-3.41	-3.121	29.34	1.16	-2.619	37.59	1.08	1.63	84.37	0.69	1.97	83.35	4.70	8.62	97.85
	Method 3	-3.41	-6.00	23.06	1.16	-5.00	29.29	1.08	-3.00	76.04	0.69	-1.00	69.05	4.70	6.00	96.58
	Simulation	1.67		9.60	1.29		3.50	8.55		99.50	3.49		68.80	8.74		99.90
Time 7	Method 1	-6.21	-6.602	4.29	10.00	5.00	84.73	-2.09	-4.156	16.24	-5.28	-7.199	3.79	0.63	0.71	56.17
	Method 2	-6.21	-6.602	5.32	10.00	5.00	95.95	-2.09	-4.156	25.36	-5.28	-7.199	4.77	0.63	0.71	56.69
	Method 3	-6.21	-9.00	4.66	10.00	5.00	86.57	-2.09	-6.00	35.24	-5.28	-8.00	5.17	0.63	0.00	56.61
	Simulation	0.00		0.00	5.00		100.00	1.00		0.20	0.00		0.00	1.39		57.00

Mr. Parker E. Bradley's report was not available at the time of publication.